Interner Bericht

Shadow-based matching for robust absolute localization during lunar landings

Hannah Kaufmann

DLR

Institut für Robotik und Mechatronik

BJ.: 2014

IB.Nr.: 572-2014/02

MASTERARBEIT

SHADOW-BASED MATCHING FOR ROBUST ABSOLUTE LOCALIZATION DURING LUNAR LANDINGS

Freigabe:	Der Bearbeiter:	Unterschriften
	Hannah Kaufmann	
	Betreuer:	
	Martin Lingenauber	
	Der Institutsdirektor	
	Dr. Alin Albu-Schäffer	

Dieser Bericht enthält 128 Seiten, 99 Abbildungen und 7 Tabellen

Ort: Oberpfaffenhofen	Datum:	Bearbeiter:	Zeichen:

Technische Universität Berlin
Faculty VI Planning, Building, Environment
Institute of Geodesy and Geoinformation Science

Master thesis

Shadow-based matching for robust absolute localization during lunar landings

Hannah Kaufmann
Matr.-nr.: 344213

Professor: Prof. Dr.-Ing. O.Hellwich
Technische Universität Berlin
Faculty IV - Electrical Engineering and Computer Science
Department of Computer Engineering and Microelectronics
Computer Vision and Remote Sensing Laboratory
Supervisor: Dipl.-Ing. M.Lingenauber
German Aerospace Center (DLR)
Institute of Robotics and Mechatronics
Department of Perception and Cognition

27th January 2014

Eidesstattliche Erklärung

Hiermit versichere ich an Eides statt, dass ich die vorliegende Masterthesis ohne fremde Hilfe angefertigt und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe. Alle Teile, die wörtlich oder sinngemäß einer Veröffentlichung entstammen, sind als solche kenntlich gemacht.

Ort, Datum	Unterschrift

Abstract

The exploration of the moon can give insights to the evolution and history of the solar system. Future lunar exploration missions require autonomous, safe and precise landings close to scientific spots of interest. Currently the German Aerospace Center (DLR, ger.:Deutsches Zentrum für Luft- und Raumfahrt) investigates several optical localization methods and the fusion of their results within the project ATON (Autonomous Terrain based Optical Navigation system). This shall allow precise landing in a region of a couple of hundred meters around a designated landing site.
This thesis investigates an image-based, monocular method for precise position estimation during lunar landings. It is especially designed for the last phase until touchdown, which is still challenging. The image taken by the spacecraft during the descent is matched with an image of the same scene, which is preprocessed prior on earth. The position can be estimated from the correlation of both images. To correlate both images, the shadows seen in the two images, are matched.
The proposed method is evaluated in software simulation. It is shown, that the method has the potential to precisely estimate the position during lunar landings, especially during the last landing phase until touchdown. The method is mostly invariant to sun angle variations and relatively large perspective differences between the images.

Zusammenfassung

Mondmissionen werden auch künftig einen hohen Stellenwert besitzen, da die Erforschung des Mondes wichtige Informationen zum Beispiel über die Entstehung unseres Sonnensystemes liefern kann. Autonome, sichere und präzise Landungen an wissenschaftlich besonderes interessanten Stellen sind dabei von Vorteil. Aus diesem Grund erforscht das Deutsche Zentrum für Luft- und Raumfahrt (DLR) derzeit im Projekt ATON (Autonomous Terrain based Optical Navigation system) Methoden zur optischen Positionsbestimmung und deren Fusion, um eine autonome, sichere und präzise Landung, innerhalb weniger hundert Meter um eine ausgewählte Landestelle, zu erreichen.
Die vorliegende Arbeit untersucht eine bildbasierte monokulare Methode zur Positionsbestimmung während eines Landeanfluges auf den Mond, wobei vor allem die finale Landephase, welche immer noch eine Herrausforderung darstellt, als Einsatzbereich vorgesehen ist. Die Positionsbestimmung erfolgt mit Hilfe einer Korrelation der vorhandenen Schatten auf einem im Landeanflug aufgenommenen Bild mit einem auf der Erde vorprozessierten Bild derselben Szene.
Die vorgeschlagene Methode wurde mit Hilfe einer Softwaresimulation getestet. Es wurde gezeigt, dass das Verfahren das Potenzial hat, die Position während eines Landeanfluges zum Mond präzise zu bestimmen. Vor allem in der finalen Landephase ist die Methode weitestgehend invariant gegen Sonnenstandsänderungen und große perspektivische Änderungen zwischen den beiden Bildern.

Contents

List of Figures

List of Tables

List of acronyms

ACC Accuracy
ALHAT Autonomous Landing and Hazard Avoidance Technology
ATON Autonomous Terrain based Optical Navigation system
BSM Binary Shadow Matching
DIMES Descent Image Motion Estimation System
DEM Digital Elevation Model
DLR German Aerospace Center (ger. Deutsches Zentrum für Luft- und Raumfahrt)
DOI Descent Orbit Injection
DTM Digital Terrain Model
EPnP Efficient Perspective-n-Point problem
ESA European Space Agency
FN False Negative
FOV Field Of View
FP False Positive
FPGA Field Programmable Gate Array
FPR False Positive Rate
gMET Maximum Entropy Thresholding with previous gamma correction
GPS Global Positioning System
HDA Hazard Detection and Avoidance systems
IAU International Astronomical Union
IMU Inertial Measurement Unit
IQR Inter Quartile Range
LIDAR Light Detection and Ranging
LOLA Lunar Reconnaissance Orbiter Laser Altimeter
ME mean Earth/Polar Axis system
MER Mars Exploration Rovers
MET Maximum Entropy Thresholding
MSER Maximally Stable Extremal Regions
N Negative
N' Predicted non-matches
NCC Normalized Cross Correlation
OI Orthographic Iteration
OpenCV Open Source Computer Vision Library
P Positve
P' Predicted matches
PDI Powered Descent Initiate
PnP Perspective-n-Point problem
PPV Positive Predictive Value
RANSAC RANdom SAmple Consensus
SIFT Scale Invariant Feature Transform
SSD Sum of Squared Differences
SSE Sum of Squared Errors
SURF Speeded Up Robust Features
TN True Negative
TP True Positive
TPR True Positive Rate

VIBAN Visual Absolute Navigation system
VISINAV Vision-aided Inertial Navigation system
VTK Visualization Toolkit

List of symbols

$\mathbf{A}$	design matrix
$\mathbf{b}$	binary vector encapsulating the neighbourhood of a shadow
$_C$	subscript/superscript indicating the camera coordinate system
$\mathbf{c}$	shadow centroid with c=(x,y)
$d1$	distance to the second nearest neighbor
$d2$	distance to the first nearest neighbor
$_{des}$	subscript indicating the belonging to the descent-image
$d_{k,i}$	depth value of point in the camera coordinate system at iteration k
$\mathbf{F}$	design matrix error propagation
f	focal length
g	grid size
$_{geo}$	subscript indicating the belonging to the geo-image
$\mathbf{G_0}$	bottom most grid pyramid layer
$\mathbf{G_l}$	top most grid pyramid layer
h	image height
H_I	entropy of an image
H_{p_s}	entropy of the probability function of the shadow pixels
H_{p_b}	entropy of the probability function of the background pixels
$_I$	subscript/superscript indicating the image coordinate system
$\mathbf{I}_\gamma$	gamma corrected image
$\mathbf{I_{src}}$	source image
$\mathbf{I_{dst}}$	destination image
$\mathbf{J}$	Jacobian matrix
k	iteration counter pose estimation after Haralick et al. [1989]
$\mathbf{K}$	camera calibration matrix
$_L$	subscript/superscript indicating the light coordinate system
$\mathbf{l_i}$	observation in a least squares adjustment
m_{pq}	image moment
o	function expressing an object on a binary image
p_i	probability of a grey scale value
p_s	probability distribution of the shadow pixels
p_b	probability distribution of the background pixels
$\mathbf{P}$	camera projection matrix
$\mathbf{Q}$	cofactor matrix
r	search radius r, gives the extend of the neighbourhood of the Grid-Algorithm
$\mathbf{R}$	rotation matrix of the exterior orientation of a camera
s	spacing of the grid
sim	similarity of the matching
$\mathbf{t}$	translation vector of the exterior orientation of a camera
t_{MET}	threshold of the Maximum Entropy Thresholding
$t^\star_{MET}$	optimal threshold found with the Maximum Entropy Thresholding
$\mathbf{T_I^O}$	transformation from the object space to the image space

$\mathbf{T_I^I}$	transformation from one image space to another image space
$\mathbf{T_C^L}$	transformation form the camera to the light coordinate system
$\mathbf{T_a}$	affine transformation estimated with the set of matched shadow centroids
$\mathbf{T_W^C}$	transformation from the camera coordinate system to the world coordinate system
$\mathbf{T_L^C}$	transformation from the camera coordinate system to the lander coordinate system
$\mathbf{T_W^L}$	transformation from the lander coordinate system to the world coordinate system
U	unknown in a least squares adjustment
$\mathbf{v_i}$	residuals
ω	weight pose estimation after [Haralick et al., 1989]
w	image width
W	subscript/superscript indicating the world coordinate system
$\mathbf{x_W}$	point in the world coordinate system
$\mathbf{x_I}$	point in the image coordinate system
$\mathbf{x_C}$	point in the camera coordinate system
α	angle between the first and second nearest neighbor
α_{sun}	angle between a sun ray and the surface
γ	gamma value for the gamma correction
δ_m	threshold to reject possibly false matches
δ	break-off criteria for the iterative depth and pose estimation
ϵ_m	single matching error
ζ	the number of close data values required to assert that a model fits well to the data using RANSAC
η	number of data values to fit the model using RANSAC
ι	the maximum number of iterations allowed with RANSAC
κ	layers of the grid pyramid
λ	longitude
μ	expectation value
ξ	threshold to define candidate shadow centroid matches
$\mathbf{\Sigma_{LL}}$	variance covariance matrix of the observations
$\mathbf{\Sigma_{xx}}$	variance covariance matrix of the unknown parameters
$\sigma_{l_i}^2$	variances of a observation $\mathbf{l_i}$
$\sigma_{l_i l_j}$	covariances of two observations
σ_0^2	theoretical variance
τ	a threshold value for determining when a data point fits a model using RANSAC
ϕ	latitude

1 Introduction

The moon has been the target of the Apollo missions and several orbiting satellites. The topography of the moon was mapped and rock samples were returned and examined. But scientists can still gather many insights from lunar exploration. They hope to find answers to the formation and evolution of the earth moon system. Further the lunar surface records the history of the solar system since 4.5 billion years. Since its birth, the moon is target of meteorite bombardments. Due to an only tenuous atmosphere and low tectonic activities the surface was preserved. In addition, the moon can give answers to the formation of differentiated planetary bodies and help to understand rocky planets in general [see Moon-C.S.C.E. et al., 2007].

To explore the scientific spots of interest spacecrafts need to land save, autonomously and close to these spots. The so called pin-point landing [Trawny et al., 2007, Pham et al., 2009] aims to land precisely in the range of a few hundred meters of a predefined landing site. Thus, pin-point landing needs the precise location of the spacecraft at any time to continuously correct the approach.

All preceding automated missions were too inaccurate for a pin-point landing. As an example of landing accuracy the landing ellipses of all NASA mars missions are illustrated in figure 1. The landing ellipse of the first mars mission, Viking in 1976, is about 280 km[1] in semi major axis and 100 km in minor axis. The Mars Exploration Rovers (MER) Spirit and Opportunity landed with an accuracy approximately double the one of the Viking mission (i.e. 149.5km x 19km). The recent Curiosity mission in 2012 shows a significant improvement to preceding missions but still has a landing accuracy in the range of kilometers. The landing ellipse reduced to 19 km x 6.5 km.

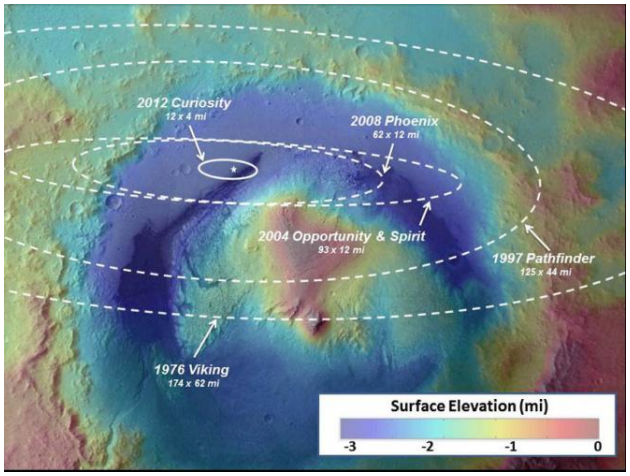


Figure 1: Illustration of landing accuracies of past mars missions. (source: [NASA, JPL-photojournal])

Vision based landing is currently the only navigation solution for pin-point landings in space, where no global position services, e.g. Global Positioning System (GPS), are available. Vision based navigation systems exploit optical sensor measurements to determine

[1] conversion into $km = mi/0.62137$

the velocity, attitude and position of a vehicle. This thesis contributes to the position estimation using optical data.

Currently the leading space agencies investigate new vision based navigation and Hazard Detection and Avoidance systems (HDA), for example the Autonomous Landing and Hazard Avoidance Technology (ALHAT) [Johnson and Montgomery, 2008, Johnson et al., 2008], Vision-aided Inertial Navigation system (VISINAV)[Trawny et al., 2007, Johnson et al., 2007, Morfopolous et al., 2011] and the Visual Absolute Navigation system (VIBAN) [Van Pham et al., 2012]. Most of the systems combine data of multiple sensors and use computer vision modules to estimate the position. The German Aerospace Center (ger. Deutsches Zentrum für Luft- und Raumfahrt) (DLR) investigates the Autonomous Terrain based Optical Navigation system (ATON), which is envisioned to provide precise localization during the descent of spacecrafts to the moon. Within the project, several different computer vision approaches to estimate the position during all lunar landing phases are investigated and combined [Theil, 2011a].

The descent of a lunar landing scenario comprises three phases [Theil, 2011a]:

- Phase 1: With the Descent Orbit Injection (DOI) the lander leaves the lunar parking orbit, which is at an altitude of approximately 100 km. The lander descends to an altitude of 10-15 km.
- Phase 2: A braking action, the so called Powered Descent Initiate (PDI) is performed. This decelerates the relative velocity with respect to the moon. At the end of phase 2 the landing site becomes visible. The altitude is approximately 1-1.5 km.
- Phase 3: The landing site is visible. The touch down is initialized.

1.1 Problem statement

ATON investigates and combines several visual navigation approaches in order to allow precise lunar landings within 200 m 3σ at designated landing sites.
The current ATON design, as shown in figure 2, combines data from a star tracker, an Inertial Measurement Unit (IMU), a camera, a laser altimeter and a LIDAR[2]. The camera images are input to different computer vision modules: Feature tracker, Crater navigation and a 3D-Matching pipeline. Each module estimates the relative or absolute position of the spacecraft. The ATON navigation filter fuses the position estimates and provides the estimated position to the lander trajectory controller [Theil and Krueger, 2011].

Feature tracking tries to detect any distinct feature in a sequence of consecutive images. This enables the estimation of the relative position during all landing phases. For a relative position determination the error increases with each estimation, hence an absolute position estimation is periodically necessary to correct for the accumulated error.

[2]Light Detection and Ranging

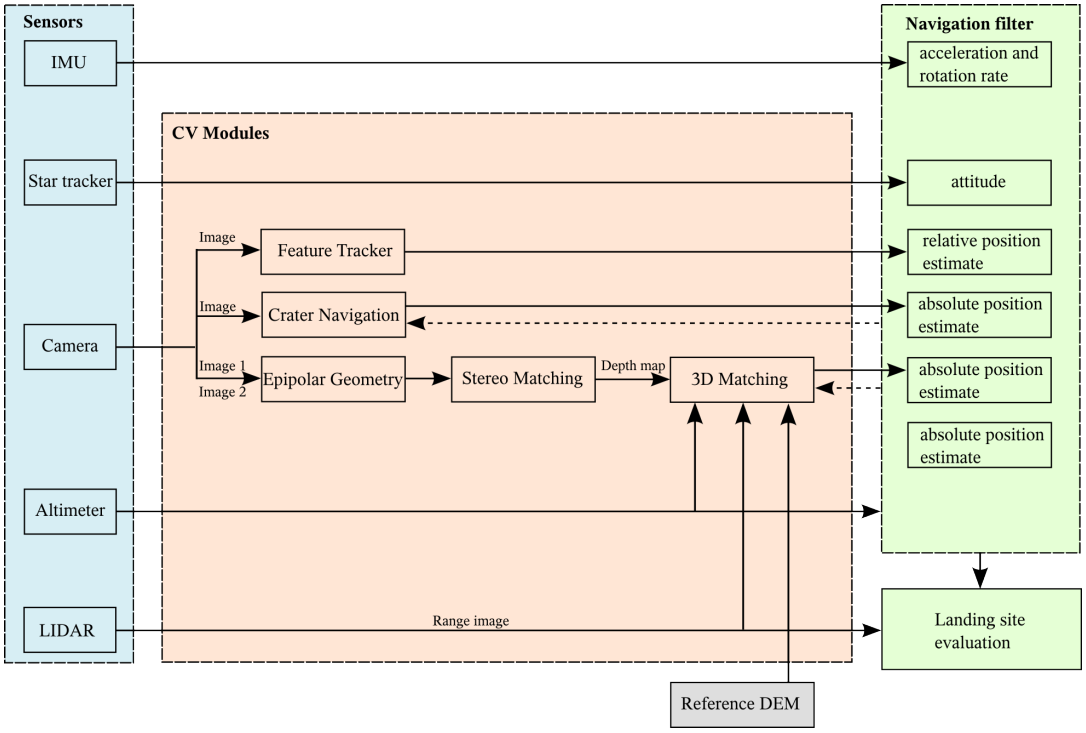


Figure 2: ATON system overview: The information of the sensors, IMU, star tracker, camera, laser altimeter and LIDAR are input to the system. The computer vision modules estimate an absolute or relative position. The navigation filter fuses the position and attitude estimates and returns the final position estimate. The range image of the LIDAR and the estimated position are used to evaluate the landing site.

Crater navigation matches camera images to a catalog of known craters and their constellation. The absolute position of the spacecraft can be estimated. The Crater navigation can be used in landing phase 1 and at the beginning of phase 2, but not in phase 3, as known craters become sparse the closer the lander gets to the lunar surface.

3D-Matching shall be capable to estimate the position during the last landing phase (last part of phase 2 and phase 3). It matches a height map acquired during the descent to a geo-referenced Digital Elevation Model (DEM) of the landing site. The height maps are either acquired with a stereo vision module, which computes dense depth images from image pairs, or a Flash LIDAR[3], a non-scanning range measurement device, that creates range images with one exposure [see Theil and Krueger, 2011].
Stereo vision based on a monocular camera (stereo vision from motion) and two stereo cameras was investigated [DLR, 2011]. Stereo vision with two cameras is not possible due to the large required baseline between the cameras. And the analysis showed, that stereo vision from motion can only be used as long as the trajectory is relatively horizontal to the surface. From approximately 2000 m on, the descent is almost vertical, therefore height maps can no longer be acquired by stereo vision with one camera. Further the use of two stereo cameras is technical not possible, as the required baseline is too long.
Instead of stereo vision the Flash LIDAR (Light Detection and Ranging) might be used, but this is still under development and its resolution (200 px x 200 px) will be relatively

[3] e.g. TigerEye 3D Flash LIDAR Camera Kit™, `http://www.advancedscientificconcepts.com/products/tigereye.html`

low compared to camera images (1024 px x 1024 px)[Theil, 2011b].
Moreover the 3D Matching module requires a preceding calculation of the epipolar geometry module and a run of the the stereo matching module, which are both computational expensive.

As shown, the current ATON system includes no approach, which is capable to estimate the absolute position in the last landing phase. An additional method, which allows to determine the absolute position of the spacecraft until or shortly before touchdown, is required.
Therefore the thesis aims to give a proof of concept of an optical navigation approach, which is able to determine the absolute position of the spacecraft in the last landing phase. The approach shall

- determine the absolute position
- be applicable until or shortly before touchdown
- be applicable to different terrain (hilly, craterous)
- allow for real-time matching and pose estimation
- use a monocular camera
- require low memory storage

1.2 State of the art

As mentioned previously, other leading space agencies currently investigate new vision based navigation and HDA systems beside ATON, e.g. ALHAT [Johnson and Montgomery, 2008, Johnson et al., 2008], VISINAV[Trawny et al., 2007, Johnson et al., 2007, Morfopolous et al., 2011] and VIBAN[Van Pham et al., 2012]. Most of the systems combine, similar to ATON, the data of multiple sensors and pose estimation modules. The estimated position and attitude of each module is the input to a navigation filter, which fuses the estimates to a global pose estimate. In the following, the focus is on the computer vision based pose estimation modules. The computer vision modules usually match features from the descent-image with a pre-calculated geo-referenced image of the scene. Among others they differ in the landmarks to be matched.

NASA investigates an Autonomous Landing and Hazard Avoidance Technology (ALHAT) for future lunar exploration missions [Johnson and Montgomery, 2008, Johnson et al., 2008]. In the context of ALHAT Cheng and Ansar [2005] try to detect craters in the descent images and to match these to known craters stored in a database.
Within the Vision-aided Inertial Navigation system (VISINAV) described by Trawny et al. [2007], Johnson et al. [2007] and Morfopolous et al. [2011], SIFT(Scale Invariant Feature Transform) features apply as landmarks. Morfopolous et al. [2011] describes the implementation of the VISINAV system in a Field Programmable Gate Array (FPGA) system using Harris Corners, which are matched by Normalized Cross Correlation (NCC) of the neighbourhood.

The Visual Absolute Navigation system (VIBAN) - a project associated with ESA - fuses a visual odometry module with the so called Landstel system [Van Pham et al., 2012, Pham et al., 2009]. Landstel, introduced by Pham et al. [2009], stores the constellation of neighbouring landmarks as a signature for each landmark. The extraction of the landmarks is done with the Harris corner detector. The extracted landmark signatures are compared with landmarks of a geo-referenced image [see Pham et al., 2009].

The previously proposed features to be matched, mainly craters and SIFT features, are exposed to different issues. Craters are sparse in some regions and the closer the spacecraft gets to the surface, the number of known craters (stored in the on board crater database) decreases.
SIFT features can be extracted from every textured image and will always be numerous. But they are not illumination invariant and their extraction is computationally expensive [Trawny et al., 2007]. Nevertheless Trawny et al. [2007] showed, that SIFT features are promising, as they can be matched shortly before touchdown.

1.3 Goal and contribution

This thesis introduces a monocular vision based pose estimation method for pin-point landing on the moon, including a proof of concept in software simulation.
Monocular vision based absolute localization for pin-point landing usually consists of three main steps [Pham et al., 2009, Trawny et al., 2007, Cheng and Ansar, 2005]:

1. Generation of a geo-referenced image of the scene (geo-image)
2. Matching landmarks of the geo-image with features of the image taken by the camera of the spacecraft during the descent (descent-image).
3. Pose estimation based on the perspective transformations between the images and the lunar surface (2D perspective - 3D point correspondence)

Figure 3 illustrates the relationship between the geo-image, descent-image and the lunar surface. By matching landmarks visible in both, the geo-image and the descent-image, the relation between the projected 2D coordinates of the descent-image and the 3D coordinates of the lunar surface is established. By solving the perspective transformation the position of the camera can be reconstructed.

The thesis introduces Binary Shadow Matching (BSM), which uses shadows as features, as shown in figure 4. The shadows on the lunar surface are detected and matched in order to estimate the absolute position of the spacecraft. The shadows on the moon have sharp boundaries, due to an only tenuous atmosphere. Therefore it is expected, that the binary shadow image can be computed and stored in a resource efficient manner. In addition, they are well distributed in rough terrain like the lunar surface. A goal of the thesis is to show, that BSM can be robust against illumination differences between geo- and descent-image. Hence, absolute localization with BSM as it is proposed in this thesis might overcome the limitations of current approaches (see section 1.2) during the final phase of landing.

Input to the BSM are two images of the scene: The image taken by the camera of the spacecraft during its descent to the surface (descent-image) and the preprocessed geo-referenced image (geo-image). The geo-referenced image is rendered from a DEM of the

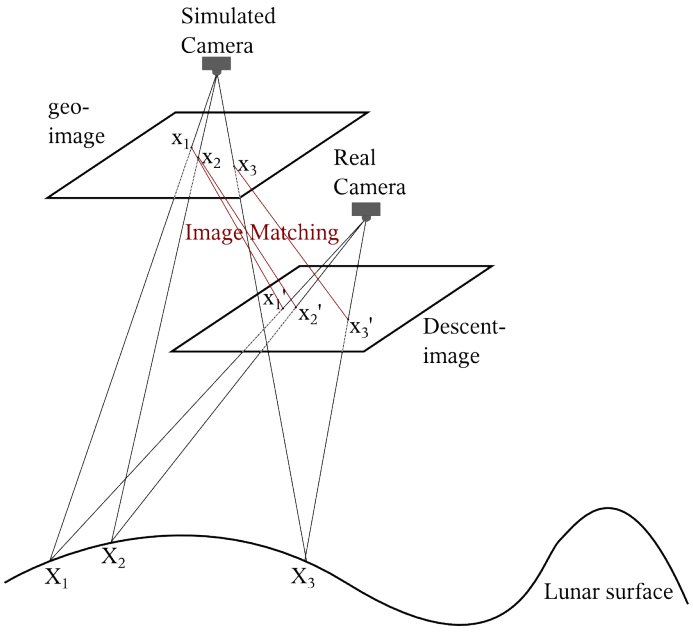
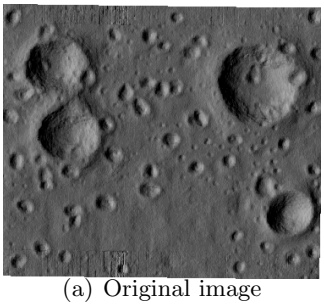


Figure 3: Concept of monocular vision based navigation. By matching the geo-image and the descent-image the relation between the 2D image coordinates of the descent image and the corresponding 3D world coordinates is established. This allows to reconstruct the position of the camera.



(a) Original image

(b) Binary shadow image

Figure 4: Binary shadow image. b) shows the binary shadow image derived from the original image a). The binary shadows apply as features in this thesis.

landing site. Output is an absolute and accurate position estimate of the spacecraft. The BSM, as illustrated in figure 5, consists of four main steps:

1. Binary shadow image generation: Two separate binary shadow images are generated, one from the geo-image and one from the descent-image.

2. Shadow extraction/description: The shadows are reduced to their centroids and stored together with information about their neighbourhood.

3. Shadow matching: The shadows from the geo-image and the descent-image are matched to correlate the images.

4. Pose estimation incl. confidence estimation: From the correlation of the images the pose of the camera is estimated. An error propagation is done to estimate the confidence of the estimated position.

The thesis includes a proof of concept of shadow based matching for robust absolute localization during lunar landing in software simulation. For this the lunar descent is simulated. Specific experiments are used to evaluate the performance and limitations of the proposed implementation in order to show that it can enhance state of the art vision based navigation systems.

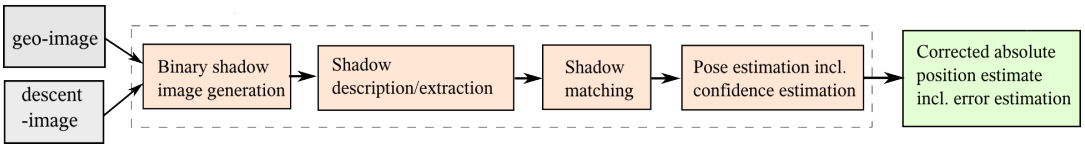


Figure 5: Binary Shadow Matching (BSM) overview: The shadows of the geo-image and the descent-image are extracted, described and stored in separate databases. The shadows are matched and the pose is estimated from the correspondence.

To the knowledge of the author shadow based matching for spacecraft navigation has not been validated so far. The only related investigation was done by Brivio et al. [1992]. They investigated the automatic selection of control-points from shadow structures in remote sensing images of high-relief terrains. Shadows were generated from the Digital Terrain Model (DTM) of the target region and were matched to the shadows of the remote sensing image.

1.4 Outline

Chapter 2 covers the related work relevant for the thesis. Chapter 3 describes the proposed shadow based matching for absolute robust navigation during lunar landings in detail. The experiments described in chapter 4 and its results (chapter 5), are used to evaluate the performance and limitations of the approach. Section 5.6 discusses the overall results of the experiments. Finally chapter 6 concludes the work and gives an outlook for future investigations.

2 Related work

The following sections introduce relevant related work for the processing steps of the BSM as shown in figure 5:

- Shadow generation (in order to pre-calculate the geo-images)
- Binary shadow image generation
- Shadow extraction, description and matching
- Pose estimation

2.1 Shadow generation

The proposed algorithm needs a binary geo-referenced image of the landing site with shadows, rendered from approximately the same view point and with light coming from a similar direction as during the descent. The geo-referenced images - geo-images - are pre-calculated on earth to save computational resources of the spacecraft. Therefore, a 3D model (used as DEM) of the landing site is rendered with shadows using shading algorithms. The light source is set to the assumed position of the sun during the descent. This leads to a geo-referenced image with equal illumination.
In the following, terms regarding shadows and shadow generation in computer graphics are presented and evaluated.

Woo et al. [1990] define a shadow as a region of darkness in an illuminated region. A region is in shadow if an object intercepts the path of light. Two types of shadows can be distinguished, hard and soft ones [see Woo et al., 1990].

Hard shadows represent a fully shadowed region, called umbra. They are caused by directional light. Diffuse light, i.e. reflected light of an incident ray on an object, causes soft shadows. Additionally to the umbra soft shadows have a penumbra, i.e. a region not fully illuminated but also not in complete darkness [Woo et al., 1990]. For example the light of the sun is reflected and refracted in the atmosphere, which leads to diffuse light. As the moon has only a tenuous atmosphere, it is sufficient to assume exclusively hard shadows on the moon. The light reflected on the surface is neglected.

Woo et al. [1990] state, that hard shadow generation is basically a binary decision. To determine the existence of a hard shadow, it is necessary to check whether an object is in the path of light or not [see Woo et al., 1990]. In the following three basic concepts for the generation of hard shadows are introduced: shadow volumes, ray tracing and z-buffer.

Shadow Volumes
Crow [1977] first described the use of shadow volumes. A shadow volume is the volume enclosed by the shadow boundaries of an object. A shadow volume is constructed by connecting a point light source with each vertex of the occluding object with a line. Two lines build a polygonal face (shadow polygon) of the shadow volume. The viewing frustum clips the planes of the shadow volume [see Crow, 1977].

The shadow polygons forming the shadow volume are divided in front facing and back facing polygons and added as hidden surfaces to the scene [Woo et al., 1990].
During the render process it is checked whether a back facing polygon or a front facing polygon is passed. Passing the front most back facing polygon means that everything in front of this polygon is in shadow until a most front facing polygon is passed [see Crow, 1977].

Ray tracing
Shadow generation can be compared with a visibility analysis from the point of view of the light source. Ray tracing makes use of this comparison. It is contributed to Appel [1968].
Woo et al. [1990] describes the two main steps of ray tracing as follows. First a ray is shot from the camera to each pixel and a ray-surface intersection analysis is performed. Second a ray from the surface intersection point is shot to the light source. The point is in shadow if the ray intersects another object on its way to the light source. Using reflection and refraction rays, also a global illumination can be considered [see Woo et al., 1990].

Z-buffer
Williams proposed the z-buffer algorithm, also known as depth-buffer or shadow map, in 1978. The z-buffer method is well-known from the render process. The z-buffer stores the depth value of each visible object. During the rendering process the image pixel is set to the color of the point, which has the smallest z-buffer value, that is the closest object to the camera at this location.
To decide which points are in shadow the z-buffer test is extended. Previous to rendering the scene from the point of view of the camera, the scene is rendered from the point of view of the light source. The depth values are stored in the shadow z-buffer, denoted as a shadow depth map [see Woo et al., 1990].

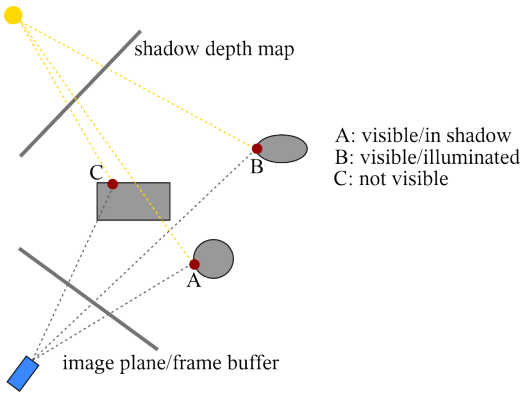


Figure 6: Z-buffer principle: For both, camera and light source a z-buffer is stored to analyze the visibility during the rendering process. A point is visible from the camera, if it has the smallest z-buffer value with respect to the camera, case A and B. Additionally, it is evaluated, if a point is the closest point to the light source, e.g. point B. If there exists a closer object with respect to the light source, the point is in shadow, as shown in case A

During the rendering, first the normal visibility z-buffer test is performed. Points x_C, y_C, z_C visible in the camera coordinate system[4], e.g. point A and B in figure 6, are transformed to the coordinate system with the light source as origin[5].

$$\begin{bmatrix} x_L \\ y_L \\ z_L \end{bmatrix} = \mathbf{T_C^L} \begin{bmatrix} x_C \\ y_C \\ z_C \end{bmatrix} \tag{1}$$

The point is in shadow, if the stored shadow z-buffer value at x_L, y_L is smaller z_L, this means a closer object exists with respect to the light source (see point A figure 6).

Discussion
The three techniques mainly differ in their computational cost and applicability.
Shadow volumes are restricted to polygonal objects. Ray tracing and the z-buffer algorithm [Williams, 1978] can be applied to arbitrary curved surfaces.

Williams [1978] states that the z-buffer algorithm has a linear cost growth, a storage complexity of *O(pq)* (see table 1 for the used notation), and a pre-processing time of *O(Enpq)* [Woo et al., 1990]. The shadow rendering complexity is constant [Woo et al., 1990]. Ray tracing needs no pre-processing and storage. The shadows are determined with a complexity equal to *O(En)* per ray shot [Woo et al., 1990]. The storage and pre-processing complexity for shadow volumes is *O(En)*. The rendering complexity is *O(En)* [Woo et al., 1990].

Due to the limitation to polygonal objects, shadow volumes are not appropriate. The ray tracing method produces very realistic shadows, but has a high shadow determination complexity. The z-buffer value is applicable to general data sets and is faster than ray tracing with negligible quality loss. Therefore, the z-buffer is used in this work to generate the geo-referenced image of the scene (geo-image) with shadows.

Table 1: Parameters of the notation of the complexity

Symbol	Definition
E	Average number of edges per polygon
n	Number of primitives in the scene
p x q	Resolution of the image in pixels

(based on Woo et al. [1990], table 1, p.14)

[4]the camera coordinate system is indicated by the subscript $_C$
[5]the light coordinate system is indicated by the subscript $_L$

2.2 Binary shadow image generation

To segment the image in shadows and background information, thresholding or clustering methods can be applied. In the following Maximum Entropy Thresholding (MET) [Kapur et al., 1985] with prior gamma correction (gMET [Cheng et al., 2001]) as well as the popular clustering algorithm k-means [MacQueen, 1967] are introduced.

K-means
K-means [MacQueen, 1967] is a well known unsupervised algorithm to cluster a set of data points. It optimizes a cost-function during assigning each image pixel to a cluster [Zaki and Meira, 2013]. This comprises mainly four steps [Zaki and Meira, 2013]:

1. choose initial cluster centroids
2. assign each image pixel to the closest cluster
3. recalculate the cluster centroids from the assigned pixels
4. iterate until a cost function, e.g. Sum of Squared Errors (SSE), with regard to the cluster centroids is below a certain threshold

MET with prior gamma correction (gMET)
Kapur et al. introduced Maximum Entropy Thresholding (MET) in 1985. MET iterates over the image histogram, in order to find the optimal threshold, which maximizes the information between image content, here the shadows, and the image background. The entropy of an image is defined as the sum of probabilities p_i of all grey scale values [Huertas et al., 2006]:

$$H_I = -\sum_{i=0}^{255} p_i \cdot \log_2(p_i) \tag{2}$$

From the original probability distribution of the grey levels two probability functions are derived with the threshold t_{MET}. The threshold, which maximizes the information between the two associated entropies, is computed iteratively. Once the optimal threshold is found, simple binary thresholding is applied to the image, in order to generate the required binary images.

Huertas et al. [2006] proved that the optimal threshold can be found with the Maximum Entropy Thresholding MET [Kapur et al., 1985] for lunar images. They apply a gamma correction previously, because the maximum entropy is limited to images with one high entropy zone, that is indicated by a bimodal histogram [Huertas et al., 2006], and call this method gMET. Figure 7 shows a possible image of the lunar surface and its histogram before and after the gamma correction. The histogram before the gamma correction shows two high entropy zones, whereas after the gamma correction only one high entropy zone exists.

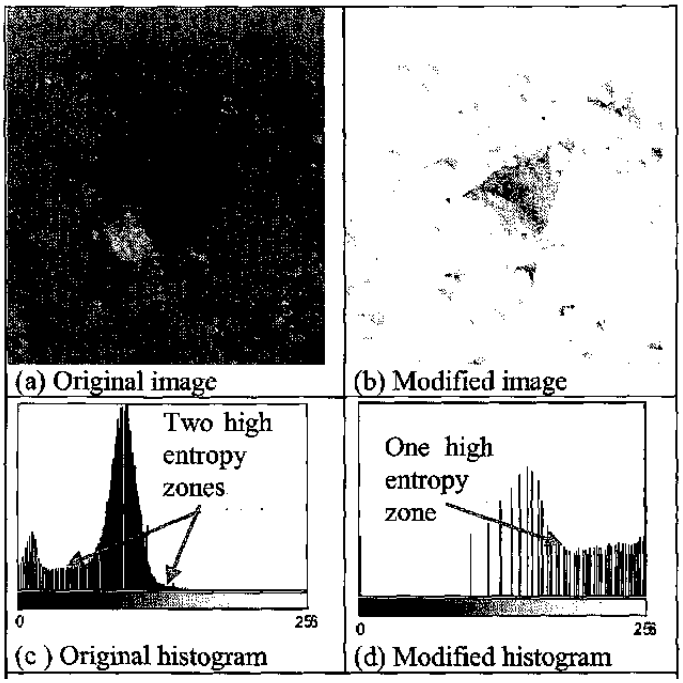


Figure 7: gMET histogram modification: The histogram (c) of the original image (a) shows two high entropy zones, histogram (d) of the modified - gamma corrected - image (b) shows only one high entropy zone.(source figure and figure caption [figure 5 Huertas et al., 2006, p.6])

Discussion

Huertas et al. [2006] state, that K-means and gMET lead to similar results. They mainly differ in the number of necessary input parameters and computational time.

To apply K-means the initial number of clusters and their centroid locations have to be chosen.

Is the number of clusters not known, a number has to be assumed. However the found clustering may be split or merged in a post-processing step. Depending on the choice of the initial cluster centroids, the algorithm may converge to a local optimum instead of a global optimum [Zaki and Meira, 2013]. To solve this problem, K-means can be applied several times with different initial cluster centroids [Huertas et al., 2006]. The trial with the smallest SSE is considered as the global optimum [Huertas et al., 2006]. But this doubles the computational time for each trial [Zaki and Meira, 2013].

gMET depends only on the γ value for the gamma correction. The γ value can be found empirically and the MET itself is independent of initial values. Huertas et al. [2006] proved that gMET is faster than K-means. Therefore gMET is used to generate the binary image in this work.

2.3 Extraction, description and matching of shadows

The matching between the geo-image and the descent-image is the central part of the BSM. This section introduces the term image matching and its fundamental concepts and strategies. It examines the relevant related work in the field of binary feature matching and relational matching.

2.3.1 Overview

Matching establishes the correspondence between either multiple overlapping images, or images and 3D objects [Paar et al., 2001]. This is denoted as image space - image space matching and object space - image space matching. Figure 8 illustrates the two correspondence problems: image space - image space $\mathbf{T_I^I}$ and object space - image space $\mathbf{T_I^O}$ (superscript = source system; subscript = destination system).

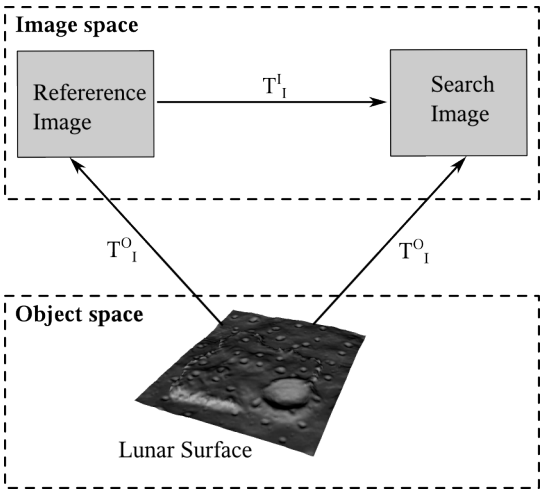


Figure 8: Matching concepts: The transformation $\mathbf{T_I^I}$ maps the reference image to the search image. The transformation $\mathbf{T_I^O}$ allows object reconstruction.(based on [Paar et al., 2001, figure 14.1,p.394])

Object space - image space matching searches the perspective transformation $\mathbf{T_I^O}$ that projects 3D object coordinates to 2D image coordinates [Paar et al., 2001]. The inverse, $\mathbf{T_I^{O-1}}$ re-projects the image coordinates to object coordinates [Paar et al., 2001].
The relation between object and image space enables the 3D reconstruction of objects or their location [Paar et al., 2001]. This includes also the localization of the camera position, from which the image was taken. The relation between object and image space is exploited in the pose estimation process described in section 2.4 and 3.5.

Image space - image space matching describes the task of establishing a correspondence between two or multiple overlapping images [Paar et al., 2001]. By exploiting the similarity of the pixels of the reference and search image, a mapping function $\mathbf{T_I^I}$ can be found. $\mathbf{T_I^I}$ maps the search image to the reference image [see Paar et al., 2001].
The mapping function $\mathbf{T_I^I}$ that relates the descent image (search image) and the geo-image (reference image) is estimated in the shadow matching process described in section 3.3-3.4.

2.3.2 Image matching

Paar et al. [2001] divides image matching into the three categories: area based matching, feature based matching and relational matching. Gülch [1994] and Walial and Suneja [2010] only distinguish between area based matching and feature based matching. Gülch [1994] describes relational matching as feature matching with relational attributes. In this work the classification of Gülch [1994] and Walial and Suneja [2010] is applied, as the proposed shadow based matching can't be assigned clearly to either feature based or relational matching.

Area based matching matches image patches or whole images directly based on grey levels or functions of grey levels [Paar et al., 2001]. The similarity is measured by a cost function, e.g. the Sum of Squared Differences (SSD) or cross correlation (e.g. Normalized Cross Correlation (NCC)). To match binary shadows this is not appropriate, because they have no textural information. Thus, feature based matching seems to be more appropriate.

Szeliski [2010, p.208,209] divides feature based matching into three steps: feature extraction, feature description, and feature matching.

Feature extraction
Feature extraction describes the extraction of reliable features. A feature is said to be reliable, if it is likely to be found in the other image too [Tuytelaars and Mikolajczyk, 2008]. Features can be corners/points, edges/lines or regions/blobs [Gülch, 1994]. To evaluate if a feature is reliable, Tuytelaars and Mikolajczyk [2008] define the following properties:

- Repeatability: The feature should have a high probability to be detected under different viewing conditions
- Distinctiveness/informativeness: The feature should be described unambiguously, e.g. variance in intensity pattern or shape.
- Locality: The features should have locally a small extend to be invariant to occlusions or geometric deformations.
- Quantity: The features should be numerous
- Accuracy: The features should be accurately localized
- Efficiency: The features should be efficiently matchable, that means with low computational costs and time

The repeatability of features is the most important property, which is required by all applications. Depending on how large the viewing conditions vary the invariance of the feature against geometric transformation or environmental conditions is important. A feature can be invariant to translation, rotation, scaling, affine transformation, perspective transformation or illumination changes. Tuytelaars and Mikolajczyk [2008] states that distinctiveness and locality are competing properties. The smaller the extend of a feature or feature descriptor, the less distinct it is.

Feature Description
The extracted features have to be described unambiguously to increase the repeatability of a feature [Szeliski, 2010]. Features can be described by:

- *radiometric descriptors:* They are based on the grey levels or a function of these of the neighbourhood of the feature. The state of the art feature descriptors based on radiometric information are for example SIFT [Lowe, 2004], SURF [Bay et al., 2008] and MSER.
- *shape descriptors:* Zhang and Lu [2004] split shape representation methods into contour-based and region-based. Examples for contour-based shape representations and descriptors are simple shape descriptors, e.g. eccentricity, circularity, major axis of orientation [Zhang and Lu, 2004], and correlation-based shape matching, e.g. Shape Context matching [Belongie et al., 2000, Belongie and Malik, 2000, Belongie et al., 2001]. Examples for region-based shape representations and descriptors are image moments, e.g. the Hu moments [Hu, 1962], or the skeleton representation [Zhang and Lu, 2004].
- *relational descriptors:* Features are described by topological and geometrical relations. Topological relations are: parallel, intersects, top of etc.. Geometrical relations can be: angle between lines, distances between centroids, etc. [Gülch, 1994]. Topological relations are invariant to perspective transformations, whereas geometrical relations are only invariant against similarity transformations. Relations are stored in descriptor vectors or in neighbourhood graphs.

The attributes are stored in feature description vectors, which are matched instead of matching directly the detected feature. The use of feature descriptors can increase the repeatability of features [Szeliski, 2010].

Feature Matching
In order to match the feature description vectors the distance, e.g. euclidean distance, between the description vectors is calculated or the nearest neighbour is searched [Szeliski, 2010]. Szeliski [2010] states that if a simple distance measurement is used a threshold has to be set, which determines whether a feature matches or not. A feature is assumed to match, if the distance is smaller the threshold.
The performance of the matching depends on the choice of the threshold. A low threshold may increase the number of correct matches, which were rejected by the threshold. At the same time the number of false positives, i.e. incorrect matches, decreases, which is desirable. Contrary a high threshold may increase the number of correct matches and the number of false matches [see Szeliski, 2010].

A nearest neighbour search matches always the nearest neighbour. No proposed match is rejected. There are only correct and incorrect matches. However, to decrease the number of incorrect matches often a threshold is applied, too [Szeliski, 2010].

To increase the efficiency of the matching process the feature descriptors can be stored in tree structures, such as the kd-tree [Szeliski, 2010].

The approach proposed in this thesis uses binary shadows as features. These have no textural information. Thus the popular descriptors, e.g. SIFT or SURF cannot be applied. Appropriate to describe binary shadows are either shape or relational descriptors.
However, an analysis showed that the morphology of the shape is not stable. The shape changes due to sun angle variations and strong perspective view point changes. It was tested to describe and match the shape with Hu moments, with the distance transform or with simple shape descriptors, such as area, orientation or aspect ratio. As the descriptions were ambiguous, the shadows could not be matched correctly. Further more, the appearance of the simulated shadows might differ from the real shadows.
Describing the shadows by their relationship to their neighbourhood seems to be appropriate. Thus a selection of appropriate relational matching methods is introduced in the following.

2.3.3 Point pattern matching

To describe the shadows by their neighbourhood, they are reduced to their centroids. This results in a point pattern. To match point patterns, the topologic and geometric relationship to the point neighbours can be exploited. These interrelationships can be stored in either neighbourhood graphs [Paar et al., 2001] or description vectors.

Representing the point pattern as graph, reduces the problem to graph matching [Li et al., 2003]. This might get complex if additional information shall be stored in the graphs and incomplete data can cause problems [Li et al., 2003]

Storing the interrelationships, e.g. polar angles or distances, in description vectors, enables one-to-one point pattern matching. This is less complex, more intuitive and requires lower computational costs for small point patterns. It is assumed that a maximum of 100 shadow features exist in one image. Thus the one-to-one point pattern matching by interrelationship description vectors is more appropriate. In the following two matching approaches based on the description and matching of interrelations as description vectors are introduced and evaluated. Both were proposed in the context of star constellation matching as it is used for star trackers.

Constellation-algorithm
Figure 9 illustrates the principle of the the Constellation-algorithm proposed by Liebe in 1993.

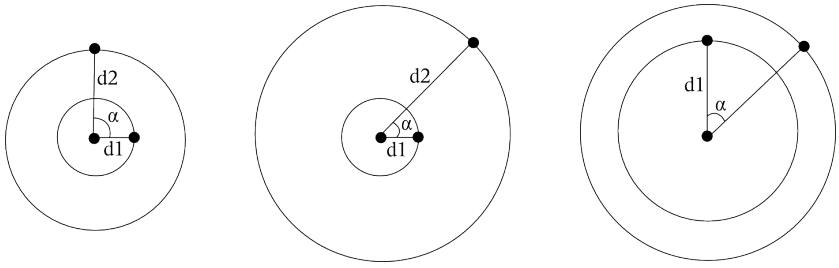


Figure 9: Constellation algorithm: The angular distances $d1$ and $d2$ to the two nearest neighbours, as well as the angle α between the two nearest neighbours describe a point. To handle deletion and addition of points the constellations of all combinations of nearest and second nearest neighbour in a certain radius are stored.(based on [Liebe, 1993, figure 10 , p.35])

A point is described with the angular distance to the first and second nearest neighbour, as well as the angle between the first and second neighbour [Liebe, 1993]. Because points may be added or deleted in one image, Liebe [1993] proposes to store all combinations of nearest neighbours in a certain radius.

Grid-algorithm

The Grid-algorithm [Padgett et al., 1997] extends the approach introduced by Liebe [1993]. The Grid-algorithm uses all neighbours and not only the two closest to describe a point. The algorithm comprises five steps [Padgett et al., 1997]:

1. Find all points within radius r of a point (figure 10 a)
2. Orient a grid, whose coordinate system has its origin at the currently observed point. The x-axis is in the direction of the closest neighbour. This enforces rotation invariance. (figure 10 b-c)
3. Project the points on a grid, whose resolution is usually much lower than that of the camera image. Each cell containing a point is assigned a 1 and empty cells a 0. (figure 10 d).
4. Derive a binary vector b from the grid. Starting from the upper left corner the grid values are stored in a vector. This results for figure 10d in:
   $\mathbf{b}$ = [000001010000000000100000...000000000000]
5. Do a binary comparison. The most similar binary vectors represent the best match.

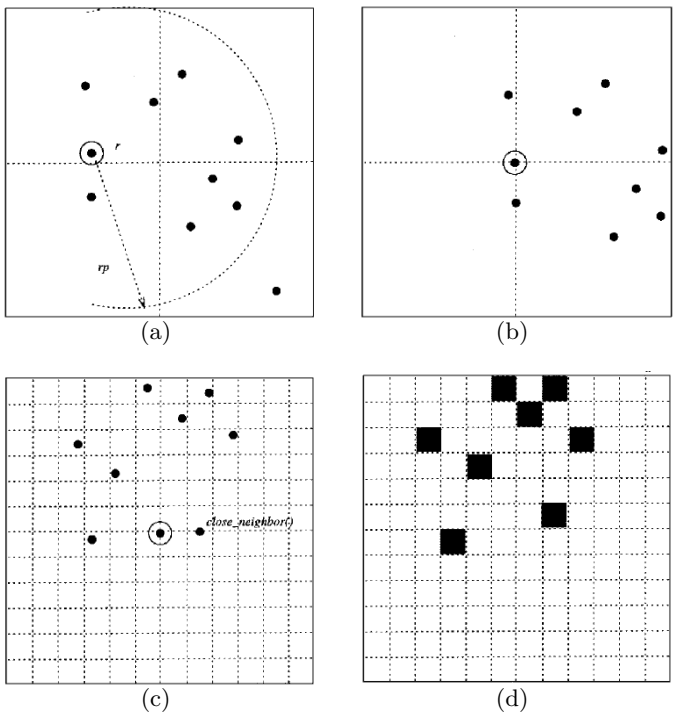


Figure 10: Grid-algorithm: a) find all stars in radius r b) set up a coordinate system with the point to be explored at the origin. c) orient a grid with the x-axis in the direction to the closest neighbour, with much lower resolution than the sensor. d) project the points on the grid: assign a 0 if the gird cell is empty, else assign 1. (figure based on [Padgett et al., 1997, figure 4, p. 263])

Discussion
The two introduced descriptors mainly differ in their informativeness, data storage and matching time. The constellation algorithm stores only interrelationship information to the first and second neighbour. While the grid algorithm encapsulates information of the whole neighbourhood in a certain radius. Thus the descriptor of the grid algorithm is usually more informative. Because the points are projected on a grid with lower resolution than the image, the descriptor is less accurate.

The descriptors of the constellation algorithm are stored in a list. Each list entry consists of 3 double values. For each shadow at least one entry is stored. To handle the addition and deletion of points, for each point several entries have to be stored. This increases the required memory storage in ordinary. The required memory for each point is not fixed, but can vary.
The descriptors of the grid algorithm are stored as binary vectors. The binary vectors are of equal size for each point. Therefore, the required memory storage is constant. Further the descriptor encapsulates a lot of information and can handle the addition and deletion of points up to a certain degree.

Both algorithms were evaluated in software simulation. Although several combinations of nearest neighbours were stored, the constellation algorithm could not handle most cases of addition and deletion of shadows. The grid algorithm proved to be faster and needs less memory storage. Further the grid algorithm lead to better matching results. The addition and deletion of shadows could be handled well. Thus, an extension of the gird algorithm was implemented to match the shadows, as detailed in section 3.4.

2.4 Pose estimation

The estimation of the absolute position of the lunar lander during the descent is the goal of the proposed method. The matching provides the correlation of the 2D image coordinates of the geo-image and of the descent-image. This further gives the corresponding 3D world coordinates of the 2D image coordinates of the descent-image. To estimate the position of the lander the 2D perspective projection - 3D problem, also known as Perspective-n-Point problem (PnP) has to be solved [Lepetit et al., 2007].

In the literature, the pose estimation problem is extensively reported [Dhome et al., 1989, Haralick et al., 1991, Horaud et al., 1989, Lepetit et al., 2007, Fiore, 2001, Ansar and Daniilidis, 2003, Long and Zhongdan, 1999]. The approaches can be divided in iterative and non-iterative algorithms.

Non-iterative algorithms
Non-iterative algorithms are further divided into algorithms using a fixed number of points - usually the minimum number of required points, that is 3 to 4 [Lepetit et al., 2007], or an arbitrary number of points. Non-iterative algorithms using a fixed number of points, e.g [Dhome et al., 1989], [Haralick et al., 1991], [Horaud et al., 1989], [Fischler and Bolles, 1981], are sensitive to noise and thus lead to non stable results [Lepetit et al., 2007]. Non-iterative algorithms using an arbitrary number of points were proposed by Lepetit et al. [2007], Fiore [2001], Ansar and Daniilidis [2003], Long and Zhongdan [1999].

Lepetit et al. [2007] introduced the Efficient Perspective-n-Point problem (EPnP). They reduce the problem to the estimation of four virtual control points. Each world coordinate point is expressed by the weighted sum of the control points [Lepetit et al., 2007]. For general configurations four non-coplanar control points are necessary, for planar configuration three. Lepetit et al. [2007] states, that the EPnP is more accurate than other non-iterative algorithms with even a lower complexity and shorter run time.

Iterative algorithms
Iterative algorithms use an arbitrary number of points. They are more accurate, but also slower than non-iterative algorithms. Two examples for iterative algorithms are the Orthographic Iteration (OI) proposed by Lu et al. [2000] and least squares with iterative depth adjustment proposed by Haralick et al. [1989].

The OI minimizes the object-space collinearity error [Lu et al., 2000], whereas Haralick et al. [1989] minimize the re-projection error. This is done by solving the projection equation with least squares. As the depth of the image coordinates is unknown this is done iteratively. After each iteration the depth is refined.

Discussion
Non-iterative and iterative methods to estimate the absolute orientation of a camera mainly differ in their robustness, computation time and accuracy. Iterative algorithms are more robust than non-iterative algorithms using a fixed number of points, but they are unstable due to noise [Lu et al., 2000]. The success of iterative algorithms relies on good initial values, in order to avoid convergence to a local minima [Lepetit et al., 2007]. Naturally, non-iterative algorithms are faster than iterative algorithms [Lepetit et al., 2007].
However, iterative algorithms are more accurate than non-iterative algorithms. But Lepetit et al. [2007] state, that the EPnP, followed by a gauss-newton optimization, leads to as accurate results as iterative algorithms, like the OI [Lu et al., 2000].

As this work is a proof of concept, the real time capability is out of scope. Hence the accuracy is graded higher than the speed and therefore an iterative algorithm was chosen. Because the attitude of the spacecraft can be accurately estimated with the star tracker only the camera position is estimated.
Although the approach of Haralick et al. [1989] leads to accurate results it is unregarded in literature. The algorithm of Haralick et al. [1989] was chosen for this first implementation.

3 Binary Shadow Matching

Figure 5 on page 7 showed the general concept of the Binary Shadow Matching (BSM). Inputs are the image taken by the camera of the spacecraft during the descent (descent-image) and a pre-processed geo-referenced image of the same scene (geo-image). By matching those images, the correlation of both images as well as the correlation between the 3D world coordinates and the 2D image coordinates of the descent-image is established. By solving the projection equation for the camera position, the absolute position of the spacecraft is estimated.

In the following the proposed BSM for spacecraft localization during lunar landings and its implementation is explained in detail. The sections are based on the related work introduced in chapter 2.

In the following all symbols representing a vector or a matrix are bold. Further vectors are symbolized with a small letter and matrices with a capital letter.

3.1 BSM overview

Figure 11 illustrates the data flow of the overall system in detail. It consists of a pre-flight and an in-flight part.

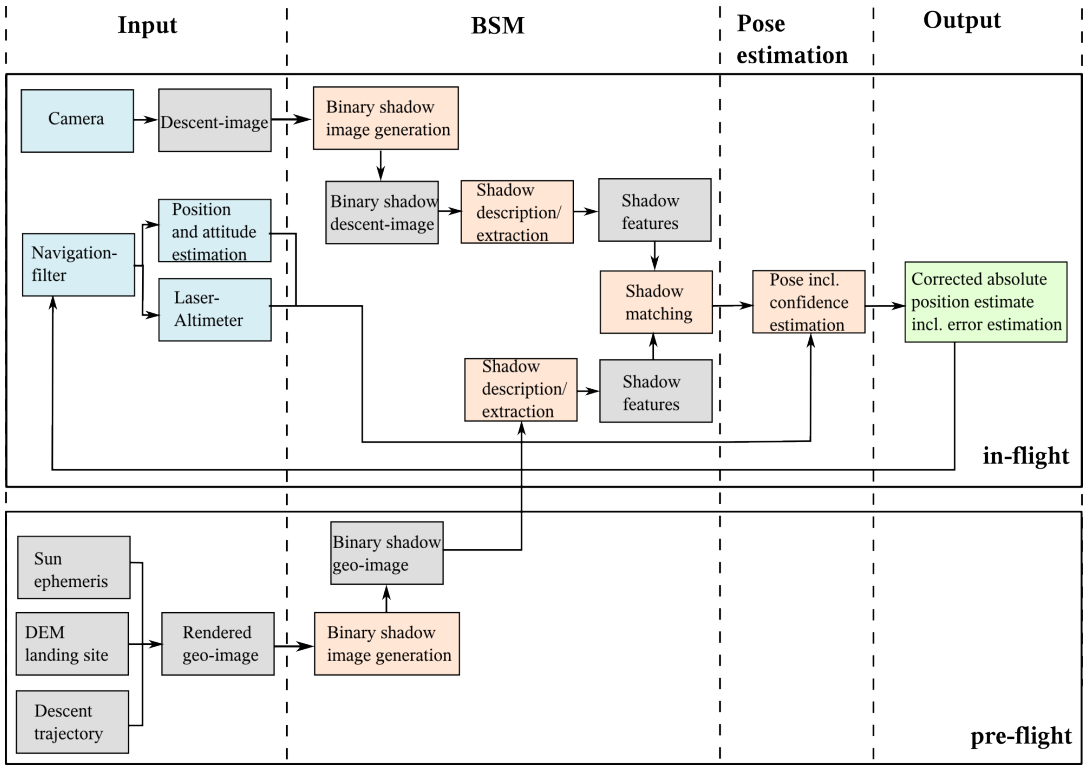


Figure 11: Processing overview of the BSM: The shadow based localization system consists of a pre-flight and in-flight part. In the pre-flight part the binary shadow geo-images are pre-processed. The in-flight part generates the binary shadow descent-image and extracts and describes the shadows of both binary shadow images. The shadows are matched and the pose is estimated.

The pre-flight processing stage generates the geo-images from the DEM of the landing site, from the descent trajectory and from the position of the sun, i.e. the ephemeris. First the geo-referenced image of the landing site, the geo-image, is rendered, second the binary shadow geo-image is derived from the geo-image.

Input to the in-flight part are the descent-image of the camera of the spacecraft, the distance measurement of a laser altimeter as well as the last position estimate provided by the navigation filter. With gMET and binary thresholding the binary shadow descent-image is generated. The shadows of both images, binary shadow geo-image and binary shadow descent-image, are reduced to their centroids and described by their neighbourhood. The resulting shadow features are matched. This correlates the geo-image with the descent-image and establishes the correspondence between the 2D image coordinates of the descent-image with the 3D coordinates of the lunar surface. The correspondence is used to solve the projection equation for the camera position, respectively the position of the spacecraft. The position estimate including a confidence estimate is passed to the navigation filter in future.

In the following, first the in-flight part of the proposed shadow based matching is described in detail. For this, it is assumed that the binary shadow geo-images are already available. Second, the generation and processing of the geo-images in the pre-flight part is explained in section 3.7, as the binary shadow image generation is equal to the one in the in-flight part.

3.2 Binary shadow image generation

Binary thresholding is applied to generate the binary shadow images. The optimal threshold, which maximizes the information between the shadow pixels and the image background, is autonomously found using the method gMET [Huertas et al., 2006](see section 2.2).

As proposed by Huertas et al. [2006] first a gamma correction is applied to the image, in order to enforce a bimodal image histogram. The gamma corrected image $\mathbf{I}_\gamma$ adds to the original intensities.

$$\mathbf{I_g} = \mathbf{I} + \mathbf{I}_\gamma \tag{3}$$

Second the optimal threshold is found with MET, as proposed by Kapur et al. [1985]. With a threshold t_{MET} two probability functions are derived from the original probability distribution of the gray-levels of the gamma corrected image. One represents the shadow pixels p_s and the other represents the background pixels p_b [Kapur et al., 1985].

$$p_s = \sum_{i=0}^{t_{MET}} p_i \tag{4}$$

$$p_b = \sum_{i=t_{MET}}^{255} p_i \tag{5}$$

The associated entropies H_{p_s}, H_{p_b} are:

$$H_{p_s}(t) = -\sum_{i=0}^{t_{MET}} \frac{p_i}{p_s} \cdot \log_2\left(\frac{p_i}{p_s}\right) \tag{6}$$

$$H_{p_b}(t) = -\sum_{i=t_{MET}+1}^{255} \frac{p_i}{p_b} \cdot \log_2\left(\frac{p_i}{p_b}\right) \tag{7}$$

Let the sum of both entropies be the information between the shadow pixels and the background.

$$t^{\star}_{MET} = \text{ Argmax}\{H_{p_s}(t_{MET}) + H_{p_b}(t_{MET})\} \tag{8}$$

The threshold $t^{\star}_{MET}$ that maximizes equation 8 is than the optimal threshold [see Kapur et al., 1985]. Once the optimal threshold $t^{\star}_{MET}$ is found, simple binary thresholding is applied to the source image $\mathbf{I_{src}}$, i.e. the original descent-image or the original geo-image. Let $\mathbf{I_{src}}(x, y)$ be the intensity value at the pixel location (x, y) of the source image and $\mathbf{I_{dst}}(x, y)$ the intensity value at the same pixel (x, y) in the destination image, it holds [6]

$$\mathbf{I_{dst}}(x, y) = \begin{cases} 1 \text{ if } \mathbf{I_{src}}(x, y) \geq t \\ 0 \text{ otherwise} \end{cases} \tag{9}$$

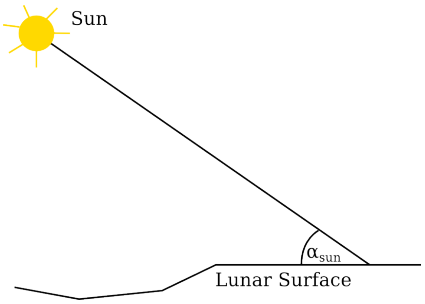
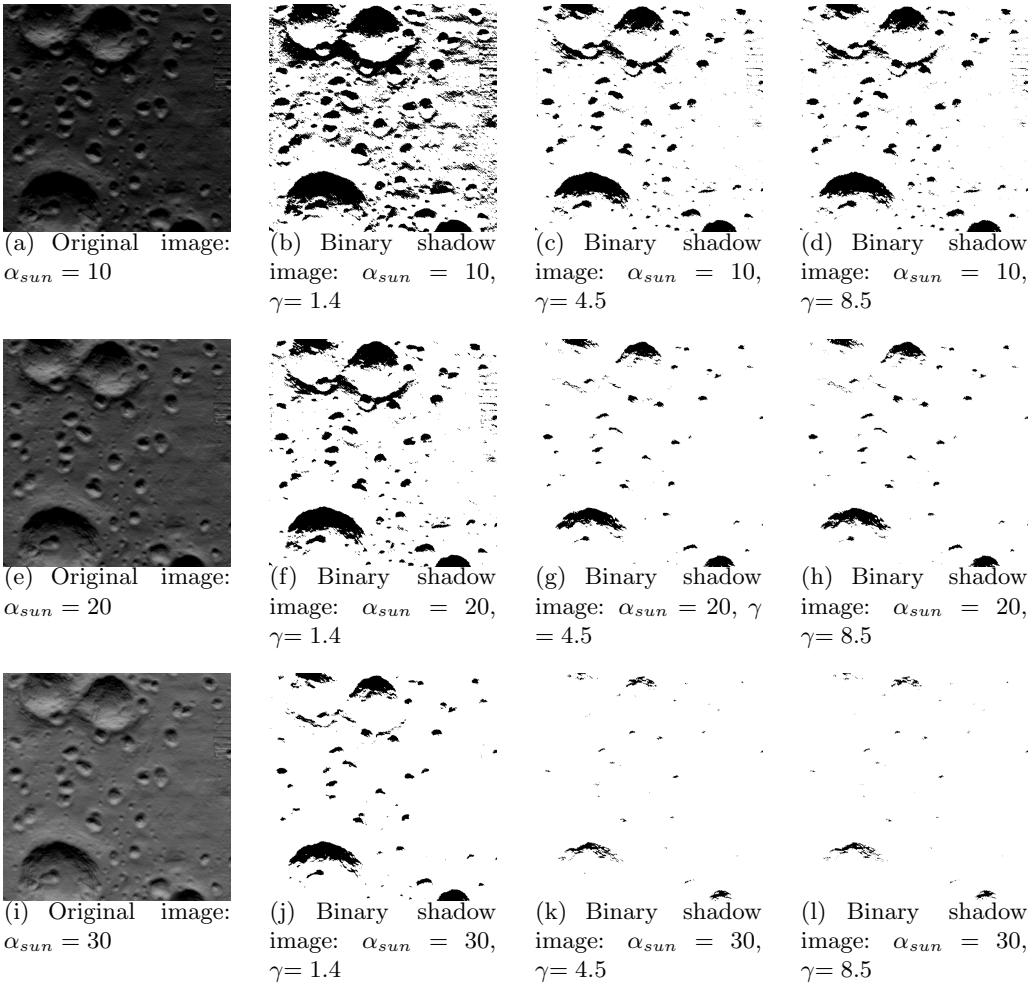


Figure 12: Sun angle α_{sun}: The sun angle is defined as the angle between the path of light and the lunar surface

As explained in section 2.2 gMET depends only on the γ value of the gamma correction. gMET is applied to different images of the lunar surface. The sun angle α_{sun} (figure 12) is varied as well as the γ value.

Figure 13 presents the results for one test image. A visual inspection performed on several different images showed that a gamma value of 1.4 leads to the best results. A higher gamma value leads to good results for small sun angles. For high sun angles the shadows are not well visible. It can be observed, that the results for a gamma value of γ= 4.5 and γ= 8.5 are quite similar. That indicates, that the small variations result in large variations for small gamma values. For large gamma values the differences are smaller.

[6] see OpenCV documentation: http://docs.opencv.org/doc/tutorials/imgproc/threshold/threshold.html



(a) Original image: $\alpha_{sun} = 10$
(b) Binary shadow image: $\alpha_{sun} = 10$, $\gamma = 1.4$
(c) Binary shadow image: $\alpha_{sun} = 10$, $\gamma = 4.5$
(d) Binary shadow image: $\alpha_{sun} = 10$, $\gamma = 8.5$
(e) Original image: $\alpha_{sun} = 20$
(f) Binary shadow image: $\alpha_{sun} = 20$, $\gamma = 1.4$
(g) Binary shadow image: $\alpha_{sun} = 20$, $\gamma = 4.5$
(h) Binary shadow image: $\alpha_{sun} = 20$, $\gamma = 8.5$
(i) Original image: $\alpha_{sun} = 30$
(j) Binary shadow image: $\alpha_{sun} = 30$, $\gamma = 1.4$
(k) Binary shadow image: $\alpha_{sun} = 30$, $\gamma = 4.5$
(l) Binary shadow image: $\alpha_{sun} = 30$, $\gamma = 8.5$

Figure 13: Results of gMET for different sun angles and gamma values: Original image with different sun angles vs. resulting binary shadow images for $\gamma = 1.4$, $\gamma = 4.5$, $\gamma = 8.5$

3.3 Extraction and description of shadows

Binary shadows are so called connected components. Those are regions of adjacent pixels with equal intensity values [Szeliski, 2010]. To extract the individual shadows the connected components have to be labeled. Here the so called border following is used.
In the image a border start pixel has to be found. This, is a pixel of value 1 with a pixel of value 0 in the 4- or 8 -neighbourhood [Suzuki and Abe, 1985]. Starting from this pixel the border of the object is followed. For details on the border following algorithm please refer to Suzuki and Abe [1985].

After labelling the shadows, they have to be described unambiguously. It is common to describe binary objects by their shape. Brivio et al. [1992] propose to describe the binary shadows by simple shape invariants. They use means of the inertia ellipses of the shadows. Because they know the extrinsic and the intrinsic parameters of the camera, as well as the exposure time, they can render the geo-referenced image of the DEM with equal point of view and sun angle. This leads to approximately similar shadows. The application of binary shadow matching, as described in this thesis, is influenced by unknown view point

changes and sun angle variations. Such changes influence the shape of the shadows. An analysis of possible view point changes and sun angle variations showed, that the morphology of the shape is not stable and thus not appropriate to describe the shadows during lunar landings. As also no textural information is available, radiometric descriptors cannot be used either. An appropriate solution is to describe the shadows by interrelationships to their neighbours. For this the shadows should be represented by one unambiguous point. The center of mass of a shadow, its centroid, is scale invariant and is sufficient to represent a shadow (see figure 14 right).

The centroids are calculated using image moments. Let the following function express an object, e.g. a shadow, on a binary image [Horn, 1986]:

$$o(x,y) = \begin{cases} = 1, & \text{point in shadow} \\ = 0, & \text{point not in shadow} \end{cases} \tag{10}$$

Hu [1962] stated, that the $(p+q)^{th}$ order moments of a density distribution function, e.g. $o(x,y)$ can be expressed in terms of Riemann integrals as follows:

$$m_{pq} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x^p y^q o(x,y)dx\ dy \qquad \text{with } p,q = 0,1,2,... \tag{11}$$

The 0^{th} moment m_{00} is equal to the area of the binary object. All shadows, whose area is smaller than 10 px are deleted. They are likely to vanish at view point and sun angle variations. For the remaining shadows the centroid is calculated. The shadow centroids $\mathbf{c_i} = (c_x, c_y)$, i.e. its center of mass, is expressed by the 1^{st} moments [Horn, 1986]. The following equations apply:

$$c_x = \frac{m_{10}}{m_{00}}\ ,\ c_y = \frac{m_{01}}{m_{00}} \tag{12}$$

Figure 14 shows the feature extraction process. From the original image (left), a binary shadow image is derived (middle). The single shadows are labelled with the previously described border following. Finally the shadow representation is reduced to the shadow centroids (right), which results in a point pattern.

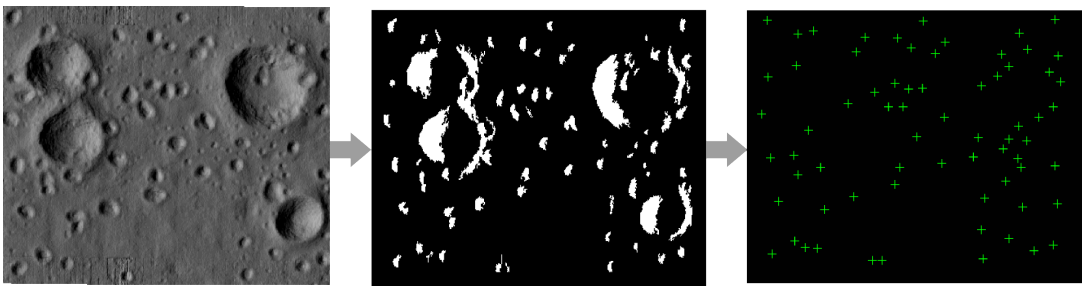


Figure 14: Shadow extraction and representation: From the original image (left) a binary shadow image (middle) is derived with Binary Thresholding. The binary shadows are reduced to their centroids (right).

Each point - shadow centroid of the point pattern contains only the information of its 2D location in the image. To describe the shadow centroids of the point pattern unambiguously, their neighbourhood can be used. Due to perspective changes and sun angle

variations between the geo-image and the descent-image shadow centroids can appear or disappear in one or the other pattern. Thus the neighbourhood of a shadow centroid differs between the two images. Hence, the descriptor should be able to handle the addition and deletion of shadow centroids in the point pattern. As shown in section 2.3 the grid algorithm [Padgett et al., 1997] is able to handle the addition and deletion of points in a point pattern. Further the descriptors have equal size for each shadow centroid and can be stored as binary vectors.

The Grid-algorithm projects the points of a point pattern on a grid with usually much lower resolution than the image. Choosing the right resolution of the grid is challenging. A small spacing of the grid is locally accurate, but not robust. Points close to the border of a grid cell are likely to jump to a neighbouring cell, which changes the descriptor. A large spacing leads to robust descriptors, but is less accurate. Therefore, the BSM uses a pyramidal implementation of the grid algorithm to describe the shadows. A pyramid of grids with different spacing is set up. This has two advantages, first a pyramidal implementation can provide a solution for the natural trade off between local accuracy and robustness, when choosing a grid resolution [see Bouguet, 2000]. Second, a pyramidal implementation can speed up the matching process. The following steps, detailed in algorithm 1, are applied to describe each shadow centroid:

1. Define the grid pyramid. Let $\mathbf{G_l}$ be the κ^{th} level of the grid pyramid with $\kappa = 0, 1, 2...l$ layers. The grid $\mathbf{G_l}$ has the lowest resolution, that is the largest spacing. The grid spacing s gives the range of pixels that projects to one cell of the grid. The grid size g_x, g_y - that is the number of grid cells - is computed with:

$$g_x = g_y = 2r/s \tag{13}$$

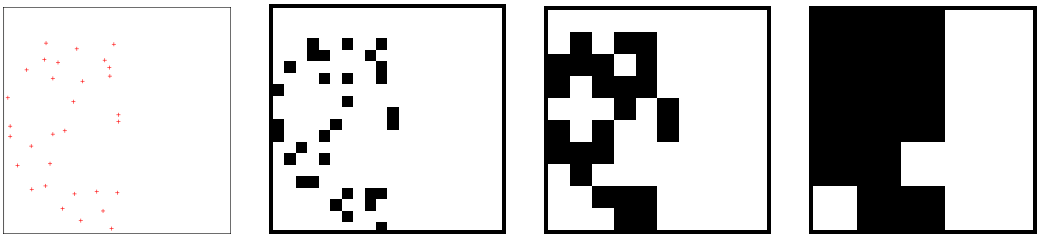
   where r is the radius in which a shadow centroid is treated as neighbour. For each layer $\kappa = 0, 1, 2, ...l$ the spacing s decreases, thus the grid size g_x, g_y increases. The spacing has to be chosen manually.

2. Find all neighbours in radius r. To speed up the neighbourhood search the shadow centroids are stored in a kd-tree. A kd-tree is a multi-dimensional search tree [Szeliski, 2010]. It enables neighbourhood searches with a complexity of[7] *O(n log n)* instead of $O(n^2)$.

3. Orient a grid, whose coordinate system has its origin at the shadow centroid $\mathbf{c} = (c_x, c_y)$.

4. Project the shadow centroids in the neighbourhood on the grid pyramid layer $\mathbf{G}_\kappa$ following algorithm 1. Each grid is reshaped as a vector $\mathbf{b_{geo}}$, respectively $\mathbf{b_{des}}$. The subscript $_{geo}$ indicates the belonging to the geo-image and the subscript $_{des}$ the belonging to the descent-image.
   For each shadow, κ binary pattern vectors, one for each gird pyramid layer, are stored. Figure 15 presents an example of the resulting binary pattern grids for three pyramid layers.

[7] scipy Cookbook/KDTree: http://wiki.scipy.org/Cookbook/KDTree



(a) original pattern (b) grid 80x80 (c) grid 40x40 (d) grid 20x20

Figure 15: Shadow description example 1: Neighbourhood grid pattern (r = 200px) for one shadow, with increasing resolution: a) original centroid pattern b) spacing = 5, c) spacing = 10, d) spacing = 20

Algorithm 1 Feature description

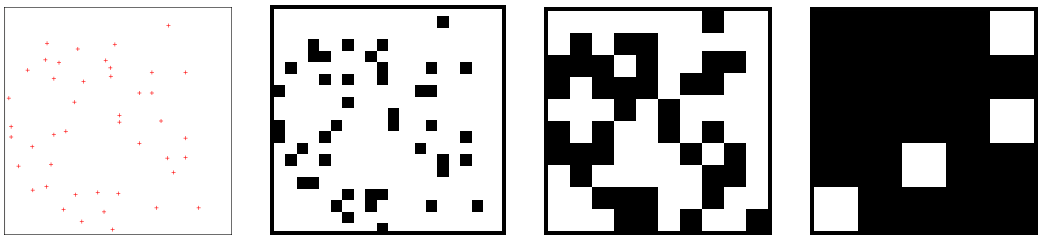
```
{Define Grid Pyramid}
κ = number of Grid layers
G_κ = Grid layer κ
s_κ = spacing of G_κ

kdtree = store shadow centroids in kd tree
for all shadows in kdtree do
  {Find neighbours}
  neighbours = query kd tree for all shadow centroids in radius r

  for G_κ do
    {Project neighbours to grid}
    for row in grid G_κ do
      for column in grid G_κ do
        cellupperBound = ((row + 1) * s_κ), ((column + 1) * s_κ)
        celllowerBound = (row * s_κ, column * s_κ)
        if shadow in cell bounds == True then
          G_κ(row, column) = 1
        else
          G_κ(row, column) = 0
        end if
      end for
    end for

    b_κ = reshape to vector (G_κ)
  end for
end for
```

3.4 Shadow Matching

For the pose estimation the correspondence of the 2D image coordinates of the descent-image and the corresponding 3D world coordinates is required. To establish this correspondence the geo-image and the descent-image have to be correlated by matching the shadow centroids of the images. The corresponding shadow centroid, is the one with the most similar descriptor. Let $\mathbf{b}_{\mathbf{geo_j},\kappa}$ be the binary pattern vector for the shadow centroid $\mathbf{c}_{\mathbf{geo_j}}$ of the geo-image and respectively $\mathbf{b}_{\mathbf{des_i},\kappa}$ the binary pattern vector for each shadow centroid $\mathbf{c}_{\mathbf{des_i}}$ of the descent-image, for each grid pyramid layer $\kappa = 1, 2, 3$.
Padgett et al. [1997] proposes to do a binary comparison. This can be expressed as [Padgett et al., 1997]

$$\sum_{i=1}^{g*g} \mathbf{b}_{\mathbf{des_i}} \wedge \mathbf{b}_{\mathbf{geo_j}} \tag{14}$$

with $\wedge$ being the logical of the two bits [Padgett et al., 1997].
Problems might occur, if a lot of shadows appear or vanish in one of the images, for example, if a shadow is at the edge of the landing site. This leads to a descent-image with usually more image content than the geo-image, which again leads to a binary pattern for a shadow of the geo-image similar to the grids presented in figure 15 and a binary pattern for the corresponding shadow in the descent-image equal to the grids presented by figure 16. It can be seen, that the patterns strongly differ. The illustrated situation cannot be handled by a simple binary comparison.



(a) original pattern (b) grid 80x80 (c) grid 40x40 (d) grid 20x20

Figure 16: Shadow description example 2: Neighbourhood grid pattern for one shadow, with increasing resolution: a) original centroid pattern b) spacing = 5, c) spacing = 10, d) spacing = 20. The shadow described by the illustrated patterns is similar to the shadow described by the patterns show by figure 15.

Instead of doing a simple binary comparison, as proposed by Padgett et al. [1997], a voting scheme, which weights the possible results of a binary comparison, is applied. Comparing the entry of two binary vectors four cases are distinguishable:

1. $\mathbf{b}_{\mathbf{geo_j}}(x) = \mathbf{b}_{\mathbf{des_i}}(x) = 1$
   In both geo- and descent pattern a shadow at similar position relative to the recently considered shadow exits.

2. $\mathbf{b}_{\mathbf{geo_j}}(x) = \mathbf{b}_{\mathbf{des_i}}(x) = 0$
   In both pattern no shadow exits at this location.

3. $\mathbf{b}_{\mathbf{geo_j}}(x) = 1$ and $\mathbf{b}_{\mathbf{des_i}}(x) = 0$
   A shadow exists in the geo pattern but not in the descent pattern. This is caused by the addition or deletion of shadows, due to large changes in the view point position or due to sun angle variations.

4. $\mathbf{b_{geo_j}}(x) = 0$ and $\mathbf{b_{des_i}}(x) = 1$
   Is the opposite of case 3. This case occurs due to two reasons. First, similar to case 3 shadows might appear or vanish, due to large changes in the view point position or due to sun angle variations. Second, the descent-image usually has more content than the geo-image, as explained previously. This case can lead to large differences in the descriptive patterns. The influence of case 4 should be decreased, to increase the probability to find the correct match of shadows at the edge of the landing site.

The occurrence of each case casts numerous votes, which up or down vote the similarity. The following votes for each case were derived empirically:

1. +10
2. +10
3. -5
4. -2

Algorithm 2 Matching the binary pattern vectors

for all shadow centroids $\mathbf{c_{geo_j}}$ with $j = 1, 2..m$ **do**
  candidateShadow$_\mathbf{i}$ $= \mathbf{c_{des_i}}$ with $i = 1, 2, ...n$
  for each pattern grid layer $\kappa = 1, 2, 3$ **do**
    for all candidateShadow$_\mathbf{i}$ do
      $sim_{\kappa_i} = \mathbf{b_{geo_j}} - \mathbf{b_{des_i}}$
      for all values of sim_{κ_i} **do**
        if $value == 0$ **then**
          $votes_{case12} += 10$
        else if $value == 1$ **then**
          $votes_{case3} -= 5$
        else if $value == -1$ **then**
          $votes_{case4} -= 2$
        end if
      end for
      $sim_{\kappa_i} = (votes_{case12} + votes_{case3} + votes_{case4})$
      if $sim_{\kappa_i} > \xi$ **then**
        candidateShadow $= \mathbf{c_{des_i}}$
      end if
    end for
  end for

  bestMatch $= \mathbf{c_{des_i}}$ with $max(sim)$
  if sim of **bestMatch** $> \delta_m$ **then**
    add **bestMatch** to set of matches
  end if
end for

Starting from the top most pyramid layer $\mathbf{G_l}$, whose grid has the lowest resolution, the similarity of the binary pattern vectors is calculated. Is the similarity larger than a certain threshold ξ the shadow centroid is stored as a candidate shadow centroid. For the next pyramid layer $\mathbf{G_{l-1}}$, with the second lowest resolution, only the binary pattern vectors of the candidate shadow centroids are compared. This is repeated until the bottom of the pyramid $\mathbf{G_0}$ is reached. The shadow centroid with the most similar binary pattern vector of this last comparison is treated as the overall best match. If the similarity of the best matching binary pattern vector is larger than a certain threshold δ_m, the match is stored in the set of matches. The approach is described in detail in algorithm 2.

Figure 17 illustrates the voting scheme with an example. The similarity of pattern 1 with pattern 2 and pattern 3 is calculated. To compute the similarity, the occurrence of each case is counted and the cast votes are summed up. As expected, the similarity of pattern 2 with pattern 1 is much higher than the similarity of pattern 3 with pattern 1.

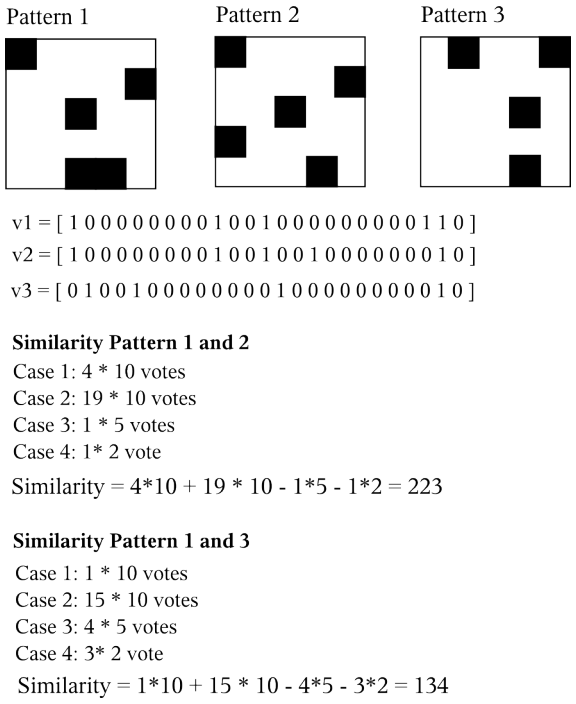


Figure 17: Shadow matching example: The similarity of pattern 1 with pattern 2 and pattern 3 is calculated. To compute the similarity, the occurrence of each case is counted and the cast votes are summed up. As expected, the similarity of pattern 2 with pattern 1 is much higher than the similarity of pattern 3 with pattern 1.

3.4.1 Outlier removal

The matches found with the matching procedure possibly contain outliers. To remove them, the best fitting affine[8] transformation $\mathbf{T_a}$, which maps the matched shadow centroids of the descent-image best to the shadow centroids of the geo-image, is found with the RANdom SAmple Consensus (RANSAC) [Fischler and Bolles, 1981]. RANSAC is well explained in the publication of Fischler and Bolles [1981] and several textbooks (e.g Szeliski

[8] affine transformation: Rotation, translation, individual scaling and shearing

[2010, p.318,319], Hartley and Zisserman [2003, p.117-121]) and not explained further.

The affine model $\mathbf{T_a}$ that fits to the most matches is estimated as follows. With η random points the affine transformation $\mathbf{T_a}$ is computed for maximum ι times. All descent-shadow centroids are transformed with the computed affine transformation. If the distance of the transformed descent-shadow centroid with $\mathbf{T_a}$ to the corresponding geo-shadow centroid is smaller than a pre-defined threshold τ, the match is considered as an inlier. And if the number of inliers is greater than ζ, the model is assumed to fit well to the data.
The affine model $\mathbf{T_a}$ that fits to the most matches is denoted as optimal $\mathbf{T_a^\star}$. The inliers of the optimal $\mathbf{T_a^\star}$ are the final set of matches. The parameters for RANSAC were found empirically.

The descent-shadow centroids of the final set of matches are further denoted as key points and are input to the pose estimation. That means key points are the descent-shadow centroids remaining after the outlier removal. Exclusively these key points are used to estimate the position.

3.4.2 Matching error

The distance of the descent-shadow centroids, transformed with the optimal $\mathbf{T_a^\star}$ to the corresponding geo-shadow centroids is a measure for the accuracy of the matching. The matching error ϵ_m of a single match is computed from the distance between the transformed 2D shadow centroid $\mathbf{c_{des_j}}$, and the corresponding 2 D shadow centroid $\mathbf{c_{geo_i}}$.

$$\epsilon_m = |\mathbf{c_{geo_j}} - (\mathbf{T_a^\star} \cdot \mathbf{c_{des_i}})| \tag{15}$$

The matching error is used to weight the image coordinates, i.e. the descent shadow centroids, at the pose estimation algorithm (see section 3.5).

3.5 Pose estimation

The matching establishes a correspondence between the 2D shadow centroids of the geo-image and of the descent-image. As figure 3 on page 6 illustrates, this correspondence further gives the corresponding 3D world points to the 2D image coordinates of the descent-image. From this 2D-3D correspondence the position of the camera and therefore the position of the spacecraft, can be estimated by solving the projection equation. The projection of 3D world points to 2D image points is expressed by [Hartley and Zisserman, 2003, p.156]:

$$\mathbf{x_I} = \mathbf{P}\mathbf{x_W} \text{ with } \mathbf{P} = \mathbf{K}[\mathbf{R}|\mathbf{t}] \tag{16}$$

where $\mathbf{I}$ is the identity matrix and $\mathbf{P}$ is the camera or projection matrix. $\mathbf{P}$ can be decomposed in the intrinsic parameters, contained in the calibration matrix $\mathbf{K}$ and extrinsic parameters, the rotation matrix $\mathbf{R}$, the translation vector $\mathbf{t}$. (for details on the camera model see appendix A).

From the projection equation the position of the camera (translation vector $\mathbf{t}$) of the spacecraft can be estimated. The orientation $\mathbf{R}$ of the camera shall not be estimated, as it is precisely (up to 0.1°) estimated by the star tracker of the ATON system. The camera position corresponds to the translation vector $\mathbf{t}$ from eq. 35. The least squares

with iterative depth adjustment as proposed by Haralick et al. [1989] and introduced in section 2.4 is used to estimate $\mathbf{t}$ and summarized in the following.

Let the 2D centroid coordinates $\mathbf{c_{des_i}}$ of the final set of matches be the n homogeneous image coordinates $\mathbf{x_{I_1}}, \mathbf{x_{I_2}}, ...\mathbf{x_{I_n}}$ with $\mathbf{x_{I_i}} = (u, v, 1)$, also denoted as key points, and $\mathbf{x_{W_1}}, \mathbf{x_{W_2}}, ..., \mathbf{x_{W_n}}$ their corresponding n homogeneous 3D world coordinates with $\mathbf{x_{W_i}} = (X_i, Y_i, Z_i, 1)$. The calibration matrix $\mathbf{K}$ as well as the rotation matrix $\mathbf{R}$ are assumed to be known. The measurement of the laser altimeter is used as an initial depth value $d_{k=0,i}$ for all image points. In the following the subscript k indicates the current number of iterations.

The position is estimated as follows:

Step 1: Equation 17 is solved with least squares [Haralick et al., 1989] (for details on least squares please refer to appendix B):

$$\epsilon_k^2 = \sum_{n=1}^{N} \omega_i ||\mathbf{K}\mathbf{R}\mathbf{x_{W_i}} + \mathbf{t_k} - d_{k,i}\mathbf{x_{I_i}}||^2 \tag{17}$$

The image points $\mathbf{x_{I_i}}$ are the observations. $\mathbf{x_{W_i}}$, $\mathbf{R}$ and $\mathbf{K}$ are fixed values and $\mathbf{t}$ is unknown. Each observation is weighted with the weights ω_i. The weights are derived from the single matching error ϵ_{m_i} as follows:

$$\omega_i = \frac{1}{\epsilon_{m_i}} \tag{18}$$

Large weights indicate an accurate matched shadow centroid or image point $\mathbf{x_I}$ respectively, and low weights indicate less accurate matched shadow centroids.

Step 2: The depth vales $d_{k,i}$ are computed with [Haralick et al., 1989]:

$$d_{k+1,i} = \left(\frac{D_{x_w}}{D_{x_c}}\right) x_{C_i(3),k}$$

where

$$\begin{aligned} D_{x_W} &= \sum_{n=1}^{N} ||\mathbf{x_{W_i}} - \bar{\mathbf{x}}_\mathbf{W}|| \\ D_{x_C} &= \sum_{n=1}^{N} ||\mathbf{x_{C_i}} - \bar{\mathbf{x}}_\mathbf{C}|| \end{aligned} \tag{19}$$

$\mathbf{x_C}$ denotes the points in the camera coordinate system, calculated with:

$$\mathbf{x_{C_i}} = \mathbf{R}\mathbf{x_{W_i}} + t_k \tag{20}$$

$\bar{\mathbf{x}}_\mathbf{W}$ and $\bar{\mathbf{x}}_\mathbf{C}$ denote the mean values of $\mathbf{x_{W_i}}$ and $\mathbf{x_{C_i}}$ respectively.

Step 3: Step 1 and 2 are repeated until the maximum difference between the depth values d_k of the $\mathrm{k^{th}}$ iteration and d_{k+1} of the $\mathrm{k^{th}}$ $^{+1}$ iteration is smaller than a certain threshold δ. The estimated translation vector $\mathrm{t^k}$ of the last iteration is used for the final pose estimate and denoted as the optimal translation vector $t^\star$.

3.6 Confidence estimation

The returned pose is an estimate containing an error. It is important to return the position estimate together with a confidence estimate. The success of the pose estimation approach mainly depends on the number of key points, their distribution and the magnitude of the error on the key points.

The error on the key points (the 2 D image coordinates $\mathbf{x_I}$ of the descent-shadow centroids) has to be propagated to the position. The error propagation proves to be difficult, as the position and the depth value depend on each other and they are iteratively estimated. Following the general law of error propagation (see appendix B and C) the error is propagated iteratively through each equation. However, this error propagation does not fully take the iterative convergence of the algorithm into account and is therefore a pessimistic guess of the error. A more realistic full error propagation is beyond the scope of this thesis.

Therefore, the following simplification is suggested. After the iterative estimation the depth is assumed to be error free. This can be assumed as the depth only changes marginally and converges after a few iterations. To estimate the error, the following equation is considered:

$$y = f(x) \rightarrow d \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbf{K}(\mathbf{R}\mathbf{x_W} + \mathbf{t}^\star) \tag{21}$$

The inverse of the function is:

$$x = f^{-1} = G(y) \rightarrow \mathbf{t}^\star = \mathbf{K}^{-1} \left(d \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} - \mathbf{K}\mathbf{R}\mathbf{x_W} \right) \tag{22}$$

The error is propagated by following the general law of error propagation (see appendix B):

$$\begin{aligned} \Delta t_x &= \sum_{i=1}^{n} \epsilon_{m_i} \sqrt{\frac{\partial G_x}{\partial u_i}^2 + \frac{\partial G_x}{v_i}^2} \\ \Delta t_y &= \sum_{i=1}^{n} \epsilon_{m_i} \sqrt{\frac{\partial G_y}{\partial u_i}^2 + \frac{\partial G_y}{v_i}^2} \\ \Delta t_z &= \sum_{i=1}^{n} \epsilon_{m_i} \sqrt{\frac{\partial G_z}{\partial u_i}^2 + \frac{\partial G_z}{v_i}^2} \end{aligned} \tag{23}$$

The error only depends on the depth d ($\approx$ distance to the surface) and the Calibration matrix $\mathbf{K}$. This is feasible, as a shift in x- and y-direction should not influence the constellation.

Besides the error propagation the reliability of the estimated pose should be considered. The reliability of a least squares adjustment can be evaluated with the parameters of the internal and external reliability known from geodetic adjustment calculations. The parameters of the internal and external reliability for geodetic networks are well explained by [Niemeier, 2008, Neitzel, 2010], please refer to the mentioned publications for detailed

information. Because the dependencies for the pose estimation after the method of Haralick et al. [1989] are not clear and the error is much larger than in geodetic networks, the interpretation of the parameters is not clear and beyond the scope of this thesis.

Instead the number of key points and their distribution can be used to estimate the reliability of the pose estimation configuration. The key points should be numerous and well distributed. Let x be all realizations of a variable, in this case all possible values of the shadow centroid coordinates c_x and c_y, and h the number of realizations. Than the uniform distribution function is given as:

$$F(x) = \frac{x}{h} \tag{24}$$

and the uniform probability function as:

$$p(x) = \frac{1}{h} \tag{25}$$

Characteristic values of the uniform distribution are the expectation value μ and the variance σ_x^2:

$$\begin{aligned} \mu &= \frac{1}{h} \cdot \sum_{i=1}^{h} x_i = \frac{1}{h} \cdot \frac{h(h+1)}{2} = \frac{h+1}{2} \\ \sigma_x^2 &= \frac{1}{h} \sum_{i=1}^{h} (x_i - \mu)^2 = \frac{1}{h} \sum (x_i - \frac{1}{h} \sum x_i)^2 = \frac{h^2 - 1}{12} \end{aligned} \tag{26}$$

As the image has a size of 1024 px in width and height, x is $x = [1, 2, 3, ..., 1024]$ for both x and y. This yields $\mu = 512$ and $\sigma_x^2 = 87381.25$.
For each key point configuration the deviation from the theoretical characteristic values σ_x^2 and μ is computed.

The result of the error propagation, as well as the number of key points and their distribution can be used to state a confidence level of the pose estimation. The results of the experiments (see section 5) show the dependencies between the number of key points and the true error of the estimated position.

3.7 Reference image generation

As described in the previous sections the descent-image is matched to a pre-calculated geo-image of a simulated scene. These geo-images are pre-processed on earth and stored in the lunar lander memory storage. In figure 11, p.20 this is denoted as the pre-flight part.
Each time step of the trajectory is stored in a state vector, which contains several parameters described in table 2. For each time step of the planned descent trajectory a geo-image is generated.

The DEM of the landing site is rendered from the perspective of the lander with the sun as light source. The camera, the sun and the DEM have to be set up in the same coordinate frame.

Table 2: State vector describing the planned descent trajectory

ID	Symbol	Description
1	$\mathbf{r_{ic}}$	Position in inertial coordinates
2	$\mathbf{v_{ic}}$	Velocity in inertial coordinates
3	$\mathbf{q_{ic,b}}$	Inertial attitude
4	$\mathbf{e_{sun,m}}$	Unit vector of sun light in moon-fixed cartesian coordinates
5	$\mathbf{r_m}$	Position of the lander in moon-fixed cartesian coordinates
6	$\mathbf{v_m}$	Velocity o the lander in moon-fixed cartesian coordinates
7	$\mathbf{q_{m,b}}$	Attitude of vehicle wrt. moon-fixed frame

The lander, on which the camera is mounted on, is placed at $\mathbf{r_m}$ and oriented by $\mathbf{q_{m,b}}$. The real intrinsics of the camera are simulated. To generate the shadows, the sun is placed at the real position corresponding to date and time. The shadows are rendered using the z-buffer algorithm [Williams, 1978] (see section 2.1). Each geo-image is passed to the binary image generation process (see section 3.2), comprising the search for an optimal threshold with gMET and applying binary thresholding.

3.8 Implementation details

The programming language Python was used to implement a prototype of the proposed method. Image processing functions taken from OpenCV[9] were applied. All numerical calculation were done with NumPy[10] and SciPy[11]. The shadow generation and rendering of the geo-image, as well as the camera simulation during the descent was done using VTK[12].

Figure 18 presents the eight main processing steps of the in-flight part in detail.

1. Binary shadow image generation (section 3.2): Input are the geo-image and descent-image.
   gMET (see section 2.2 and Kapur et al. [1985], Huertas et al. [2006]) finds the optimal threshold, which maximizes the information between shadow pixels and background of the input images. Binary thresholding is applied to create the binary shadow images of the descent-image and the geo-image.
   The binary shadow descent-image and the binary shadow geo-image are returned.

2. Shadow extraction (section 3.3) Input are the two binary shadow images.
   With the OpenCV function cvfindcontours() the binary shadows are labeled as connected components. For each shadow, i.e. each connected component, the image moments are calculated with the OpenCV function cvmoments(). From the image moments the area of the shadows and the centroids are calculated. Small binary

[9] http://opencv.org/
[10] http://www.numpy.org/
[11] http://www.scipy.org/
[12] http://www.vtk.org/

shadows are removed from the images. Two separate sets, one for each input image, of shadow centroids (shadow features) are returned.

3. Shadow description (section 3.3) Input are the set of shadow centroids.
   To describe the neighbourhood of each shadow, hence the shadow itself, it is encapsulated in a binary pattern, based on a pyramidal implementation of the grid-algorithm ([see Padgett et al., 1997] and section 2.3.3). The grids are reshaped to binary vectors and stored.

4. Shadow matching (section 3.4) Input are the binary vectors describing the shadows by their neighbourhood.
   To find the best matching shadow, a weighted binary comparison is done recursively for each pyramid layer. The matching is based on the nearest neighbour search and a set of candidate matches is returned.

5. Outlier Removal (section 3.4.1): Input is the set of matches.
   The found matches can contain outliers. To remove these, the best fitting affine transformation between the matches is estimated with the RANSAC-scheme [Fischler and Bolles, 1981]. The matches consistent with the best fitting affine model present the final set of matches.
   The final set of matches is returned. The matched centroids of the descent-image are the key-points used to estimate the position.

6. Matching error estimation (section 3.4.2): Input is the estimated best fitting affine transformation and the final set of matches.
   The matching error of a single match is defined as the euclidean distance between a shadow centroid in the geo-image and the transformed corresponding centroid of the descent-image with the estimated affine transformation.
   The matching error of each shadow centroid is returned.

7. Pose Estimation incl. confidence estimation (section 3.5 and 3.6): Input are the matched 2D centroid coordinates of the descent image (also denoted as key-points), their corresponding 3D coordinates and the matching error.
   With the final set of matches and the corresponding 3D coordinates the pose is estimated by solving the projection equation with a linear least squares with an iterative depth adjustment [Haralick et al., 1989]. The matching error is used to weight the observations, i.e. the 2D coordinates of the shadow centroids in the descent image (key-points).
   The match error, contained in the 2D centroid coordinates, is propagated to the position, following the general law of error propagation. The estimated position incl. an error estimate and a confidence estimation is returned.

The estimated position, its confidence estimation and its error estimation will be input to the ATON navigation filter in the future. The navigation filter will fuse the pose estimate with the pose and attitude estimates from the other modules of the ATON-system (see section 1.1, p. 2).

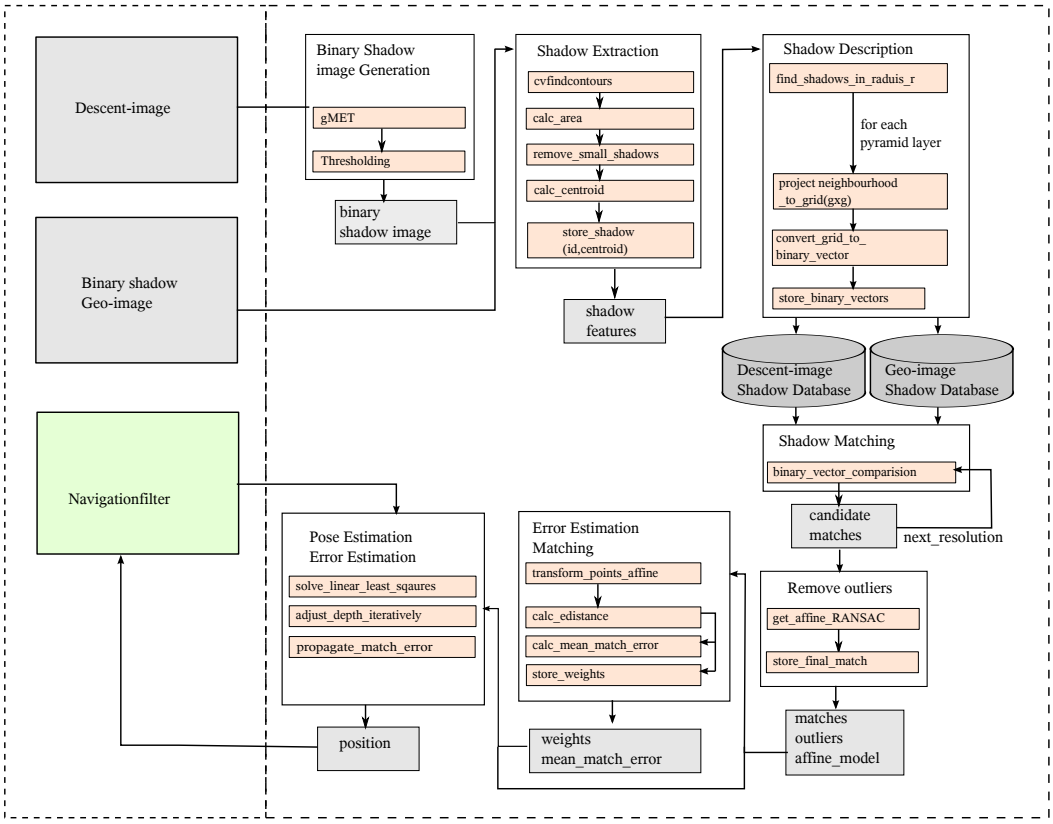


Figure 18: Detailed process overview of the BSM implementation: Input are the descent-image, the binary shadow geo-image and the last pose estimate of the navigation filter. From the descent-image the binary shadow descent-image is derived. The shadows of both binary shadow images are described. Based on their description the shadows are matched. Possible outliers are removed and the matching error is calculated. The position is estimated and returned to the navigation filter.

4 Experiments

The performance and limitations of the method are evaluated by software simulation. The geo-image and the descent-image is simulated for the descent trajectory. To evaluate the performance and behavior of the method, three experiments are applied to three landing site models. In the following, first the experimental set up and second the experiments themselves are described.

4.1 Experimental setup

This section describes the part of the experimental set up that is equal for all experiments, e.g. the definition of the coordinate systems, camera simulation, landing site models.

4.1.1 Coordinate systems

The three coordinate systems, moon-fixed coordinate system, lander coordinate system and camera coordinate system, have to be defined. All calculations are performed in the moon-fixed coordinate system. The attitude and position of the camera is given in the camera coordinate system, which is defined with respect to the lander coordinate system.

For the camera simulation each scene component has to be transformed to the moon-fixed coordinate system. Figure 19 illustrates the transformations between the coordinate systems. The superscript indicates the source system, the subscript the destination system. The moon-fixed coordinate system is denoted with the super-/subscript W for world, the lander coordinate system with the super-/subscript L and the camera coordinate system with the super-/subscript C.

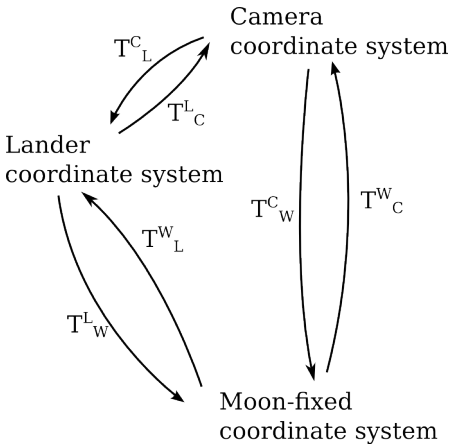


Figure 19: Overview of the coordinate systems: Transformations between the moon-fixed coordinate system (super-/subscript W), the lander coordinate system (super-/subscript L) and the camera coordinate system (super-/subscript C). The superscript denotes the source system and the subscript the destination systems.

Moon-fixed coordinate system:
The mean Earth/Polar Axis system (ME-system), as recommended by the IAU report [Archinal et al., 2010], is applied. The system is defined by its rotation axis and prime meridian, as illustrated in figure 20. The mean rotational pole defines the Z_W axis

of the ME-system. The prime meridian is equal to the mean earth direction and defines the X_W axis. For detailed information of the definition of the ME-system please refer to the IAU report [Archinal et al., 2010].

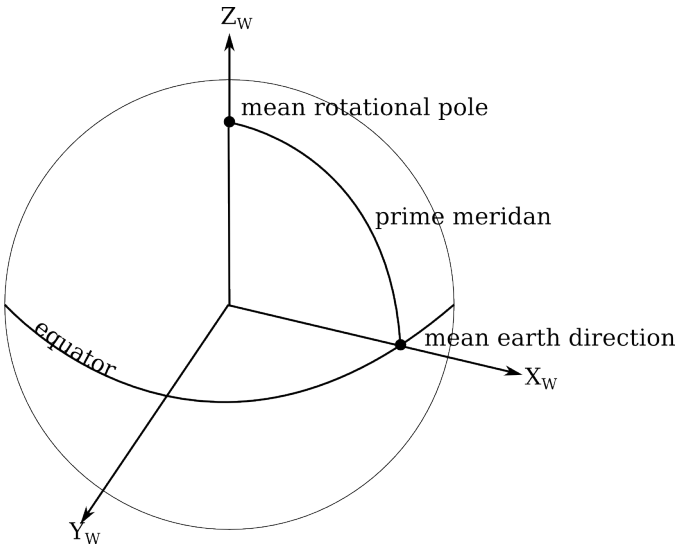


Figure 20: Moon-fixed coordinate system: The mean rotational pole defines the Z_W axis, the X_W axis is equal to the mean direction to the Earth, which also defines the prime meridian. (figure based on [Maas and Verveld, 2011, figure 2, p.7])

Lander coordinate system:
The local lander coordinate system is a right handed system fixed at the spacecraft (figure 21). The positive X_L axis points in the direction of the propulsion axis. The Y_L axis is orthogonal to the X_L axis in a way, that a rotation around the Y_L axis is equal to pitching the spacecraft. The Z_L axis completes the system [see Maas and Verveld, 2011].

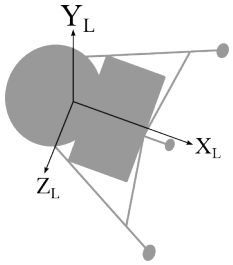
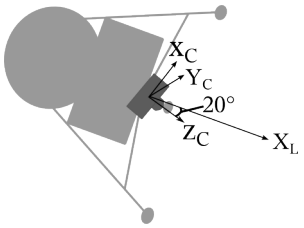


Figure 21: Lander coordinate system: Lander coordinate system fixed at the spacecraft. The X_L axis is in the direction of the propulsion axis. The Y_L axis is orthogonal to x_L, such that a rotation around Y_L is equal to pitch the spacecraft. The Z_L axis completes the system (figure based on [Maas and Verveld, 2011, figure 6, p.13])

Camera coordinate system:
With respect to the lander coordinate system the camera position is
$\mathbf{c_L} = \begin{bmatrix} -0.5\,m & 0.2\,m & 2\,m \end{bmatrix}$. The Y_C axis is rotated about half the negative opening angle (Field of View - FOV) of the camera and about 90°, thus the rotation angle is $\psi = 90° - FOV/2$ [see Maas and Verveld, 2011]. Because of the rotation about 90° the

Z_C axis is equal to the optical axis. Hence, the rotation matrix, $\mathbf{R_L^C}$ and the translation vector $\mathbf{t_L^C}$ to transform the camera coordinate system to the lander coordinate system are:

$$\mathbf{R_L^C} = \begin{bmatrix} \cos(\psi) & 0 & \sin(\psi) \\ 0 & 1 & 0 \\ -\sin(\psi) & 0 & \cos(\psi) \end{bmatrix} \text{ with } \psi = 90° - \frac{FOV}{2} \text{ with } FOV = 40° \text{and}$$
$$\mathbf{t_L^C} = \begin{bmatrix} -0.5\,m \\ 0.2\,m \\ 2\,m \end{bmatrix} \tag{27}$$



(a) Camera coordinate system

Figure 22: Camera coordinate system: The Camera coordinate system is fixed at the camera center. The Z_C axis is equal to the viewing direction. The Z_C axis is rotated about $-20°$ with respect to the lander X_L axis (figure based on [Maas and Verveld, 2011, figure 7, p.13]).

4.1.2 Camera simulation

To simulate the camera during the descent, the scene, composed of the DEM of the landing site, the sun and the lander with its camera, is set up, as illustrated in figure 23. The common coordinate system is the moon-fixed coordinate system, as defined in the section 4.1.1. All components of the scene are transformed to the moon-fixed coordinate system. The scene is rendered with the Visualization Toolkit (VTK)[13].

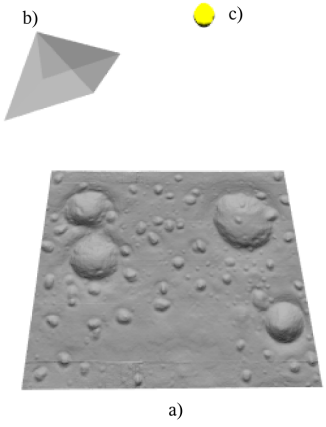


Figure 23: Simulated scene: a) lunar surface (DEM of the landing site); b) camera mounted on the lander; c) sun (directional light source)

[13] http://www.vtk.org/

The interior and exterior parameters of the camera (explanation see appendix A) define the camera parameters and its position and attitude. The exterior parameters define the position and attitude of the camera in the moon-fixed coordinate system. The camera pose is transformed to the moon-fixed coordinate system applying the transformation $\mathbf{T}_{\mathbf{W}}^{\mathbf{C}}$:

$$\mathbf{T}_{\mathbf{W}}^{\mathbf{C}} = \mathbf{T}_{\mathbf{W}}^{\mathbf{L}} \cdot \mathbf{T}_{\mathbf{L}}^{\mathbf{C}} \tag{28}$$

The interior parameters, given as the calibration matrix $\mathbf{K}$ (eq. 29), are shown in table 3.

$$\mathbf{K} = \begin{bmatrix} f & 0 & h/2 \\ 0 & f & w/2 \\ 0 & 0 & 1 \end{bmatrix} \tag{29}$$

The principal point is set at the image center $(c_x, c_y) = (w/2, h/2)$ [Szeliski, 2010].

Table 3: Camera intrinsics

Notation	Value	Description
w	1024	image size
h	1024	image size
FOV	40°	field of view
f[1]	1406	focal length

[1] with $f = \frac{W}{2}\left[\tan\frac{FOV}{2}\right]^{-1}$ [Szeliski, 2010]

Given is a 3D point cloud of the landing site. To increase the performance, the point cloud is tiled. From each tile a quadrilateral mesh is generated and rendered. The shadows casted by the terrain are rendered with the z-buffer algorithm (explanation see section 2.1).

The sun, the light source of the scene, is far away and therefore simulated as a directional light source. The state vector contains the current position of the sun $e_{sun,m}$ (see table 2 on page 34 for the state vector parameters).

4.2 DEMs of the landing sites

Three landing site models with different surface characteristics, are available to test the method. The surface characteristics strongly influence the shape, size and number of cast shadows. The models are from two different data sources. They differ in their extent and resolution.

DEMs form data products of the Kaguya Terrain Camera [Haruyama et al., 2012] and DEMs from the Lunar Reconnaissance Orbiter Laser Altimeter (LOLA) [Backer et al., 2012] are available for most parts of the moon. Because their resolution is low (approximately 7.5 m), they are denoted as low resolution DEMs in the following.

To generate high-resolution DEMs the DLR shaped different lunar surfaces by hand in a soil bin [Lingenauber et al., 2013]. The DEMs were generated from camera images of the soil bin. The soil bin has an extend of 5 m x 6 m, the DEMs were scaled up by factor 100. This results in DEMs with an extend of 500 m x 600 m. The resolution of the DEMs is 0.3 m [see Lingenauber et al., 2013]. They are denoted as high resolution DEMs in the following.

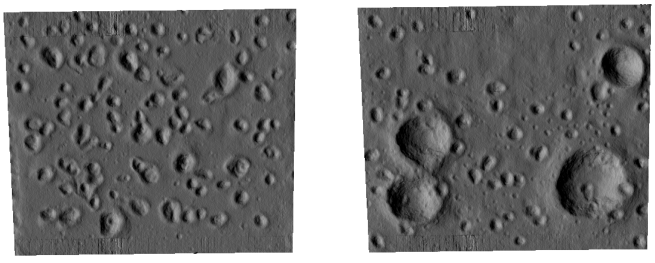
The experiments are limited due to the resolution and extend of the available DEMs.
Tests with the low resolution DEMs are restricted to an altitude of 35-12 km. The high-resolution DEMs have only a small extent of 500 m x 600 m and can be used to simulate an appropriate geo-image but not an appropriate descent-image at high altitudes. The descent image should map the landing site plus the surrounding terrain, a larger model containing the landing site model would be more appropriate. Therefore the high-resolution DEMs can be used to test at an altitude of 4 - 1 km.

The datasets of both sources have to be pre-processed. The xyz coordinates of the high resolution DEMs are in a local coordinate frame and have to be transferred to the moon-fixed coordinate system. The derivation of the transformation equations can be found in appendix D.
The low resolution DEM contains artificial noise in the height direction. This causes artificially rough terrain where non such exists. In order to reduce the noise the DEM is filtered in a pre-processing step with a box filter (kernel size of 21 x 21).

In the following the three landing sites, used to test the method, are described. Landing site 1 and 2 are high resolution DEM and landing site 3 is a low resolution DEM.

4.2.1 Landing site 1: Numerous and well distributed craters

This landing site model (figure 24 a) is a high resolution DEM. It is located at latitude $\phi = -5°$ and longitude $\lambda = 8°$. The model contains 40 to 50 randomly distributed craters, of different size. The crater diameters vary from 1 m to 5 m. This terrain is assumed to be ideal for the proposed shadow based matching, as it contains a lot of craters, that are shadow casting objects, which are well distributed. The geo-images as well as the descent-images are simulated with this model.



(a) Landing site 1 (b) Landing site 2

Figure 24: Illustration of the landing sites 1 and 2: a) Landing site 1: Terrain with numerous well distributed craters with diameters of 1 m to 5 m, high resolution 0.3 m, hand shaped in soil bin; b) Landing site 2: terrain with small and four large craters, craters are concentrated in the middle and south, high resolution 0.3 m, hand shaped in soil bin.

4.2.2 Landing site 2: Small and large craters

This landing site model (figure 24 b) is a high resolution DEM. It is located at latitude $\phi = -5°$ and longitude $\lambda = 8°$. The model contains four large craters with a diameter of 50 m to 150 m with varying depth. There are two large craters in the south east and two in the west. Randomly small craters are distributed over the model. The model has numerous craters, which are mainly concentrated in the middle and south of the model. The geo-images as well as the descent-images are simulated with this model.

4.2.3 Landing site 3: Hilly terrain with few craters

This model (figure 25) is a low resolution DEM. It is located at latitude $\phi \approx 46°$ and longitude $\lambda \approx 8°$. The terrain is hilly and contains only a few craters. The shadow casting features, hills and craters, are well distributed but sparse. With this landing site the matching of shadows caused by hills can be tested. Please note that the geo-image is generated from the model represented by figure 25(right) and the descent -image is generated from a larger model containing the landing site, figure 25(left).

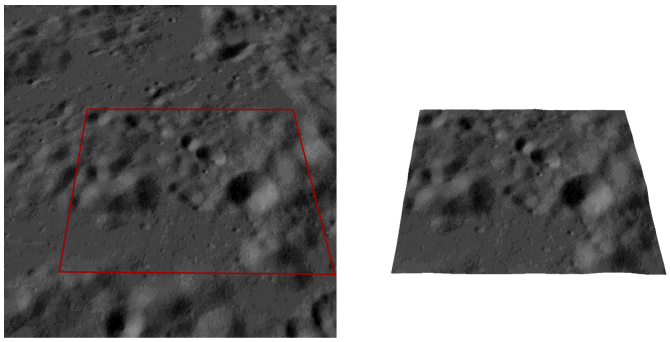


Figure 25: Illustration of landing site 3: left model for the descent-image, right model for the geo-image. Hilly terrain with few craters. Hills and craters are well distributed. The DEM was derived from Kaguya mission products and has a low resolution ($\approx 7.5m$).

Table 4 summarizes the properties of the three DEMs of the landing sites.

Table 4: Properties of the DEMs of the landing site

Name	Data source	Resolution	Extent	Location
Landing site 1	hand shaped in a soil bin	0.3 m	500 m x 600 m	$\phi = -5°$ $\lambda = 8°$
Landing site 2	hand shaped in a soil bin	0.3 m	500 m x 600 m	$\phi = -5°$ $\lambda = 8°$
Landing site 3	Kaguya Products, DEMs of LOLA[1]	$\approx$ 7.5 m	37.5 km x 56.8 km	$\phi \approx 46°$ $\lambda \approx 8°$

[1] Lunar reconnaissance orbiter laster altimeter

4.3 Algorithm parameters

The proposed algorithm requires the definition of several default input parameters. The parameters and its values presented by table 5 and table 6 are equal for all experiments. The values were mostly determined empirically. The shadow segmentation and pose estimation procedure are mostly independent of default parameters, they only depend on one parameter. The matching and outlier removal depend on several default parameters, which cannot be estimated autonomously and have to be set manually. They are valid for all experiments.

Table 5: Overview of the parameters of the BSM a)

Process	Name	Value	Description
Binary shadow generation	γ	1.4	value for the γ correction ($\mathbf{I_g} = \mathbf{I}^{\gamma}$) previous to MET[1]
Matching	$minArea$	10	all shadows having a smaller area than $minArea$ are deleted
	r	$200px$	shadow within radius r are treated as neighbours and encapsulated in the neighbourhood pattern
	l	3	number of grid pattern pyramid layers
	$s_{l=0}$	20	spacing for the 0^{th} grid pattern pyramid layer
	$s_{l=1}$	40	spacing for the 1^{st} grid pattern pyramid layer
	$s_{l=2}$	80	spacing for the 2^{nd} grid pattern pyramid layer
	$votes_{case12}$	+10	number of votes cast for each occurrence of case 1 and case 2. Case 1: in both patterns a shadow exists. Case 2: in both patterns no shadow exists.
	$votes_{case3}$	-5	number of votes cast for each occurrence of case 3. Case 3: in the geo-image exits a shadow and in the descent-image not.
	$votes_{case4}$	-2	number of votes cast for each occurrence of case 4. Case 4: in the descent-image exits a shadow and in the geo-image not.

Table 6: Overview of the parameters of the BSM b)

Process	Name	Value	Description
Outlier removal (RANSAC[2])	η	6	minimum number of data values to fit the model
	ι	1000	the maximum number of iterations allowed with RANSAC
	τ	10	a threshold value for determining when a data point fits a model
	ζ	1	the number of close data values required to assert that a model fits well to the data
Pose Estimation	δ	0.1	break-off criteria for the iterative depth and pose estimation. The algorithm stops if the difference between the depth values of the $k+1^{th}$ and k^{th}iteration is smaller δ

4.4 Experiment description

To evaluate the performance and limitations of the algorithm three experiments are applied. All experiments were applied to the three landing sites described in section 4.2.1, 4.2.2 and 4.2.3. For all the experiments the default parameters defined in table 5 apply. The experiments are applied to a section of the trajectory. The section of the trajectory is often defined by the distance to the surface. This corresponds to the measurement of the laser altimeter. It is not equal to the height over the surface. It is the distance from the spacecraft to the surface measured in direction of the optical axis of the camera.

4.4.1 Experiment 1: Robustness with respect to an initial trajectory offset

This experiment evaluates the behavior of the method with respect to an increasing initial offset of the trajectory. Hence the deviation of the true position from the planned trajectory. It is given as the distance to the assumed trajectory position. For example, if the offset is 50 m, the error on the x, y, and z coordinate is populated in such a way, that the euclidean distance between the real position and the assumed trajectory position becomes 50 m.

This experiment shall show the robustness to an initial trajectory offset. The trajectory offset leads to perspective differences between the geo-image and descent-image. This makes the binary shadow matching and the following pose estimation challenging.

The experiment is applied 5 times with increasing trajectory offset of 50 m, 100 m, 200 m, 300 m, 400 m and 500 m. For landing site 1 and 2 the experiment is repeated 50 times for each offset. For landing site 1 the experiment could only be repeated 14 times, because the processing time for the DEM was, due to its great extend, very high.

4.4.2 Experiment 2: Robustness with respect to sun angle variations

This experiment evaluates the behaviour of the method with respect to illumination changes, e.g. due to a time delay. This means the spacecraft arrives later at a position on the trajectory as expected. This causes a deviation of the sun position, which in turn leads to differently shaped shadows. Because the allowed difference of time is more important for this application, the error is given in time and not in elevation and azimuth of the sun. The experiment is applied 6 times with increasing time difference of: 5, 15, 25, 35, 45 and 50 minutes[14].

4.4.3 Experiment 3: Robustness with respect to a trajectory offset and sun angle variation

This experiments combines experiment 1 and 2. An initial trajectory offset as well as a time difference is considered. Thereby an offset up to 100 m, and a time difference up to 50 minutes is randomly generated. The experiment is performed 100 times for landing site models 1 and 2, and 13 times for landing site model 3.

4.4.4 Evaluation measurements

To evaluate and compare the results of the experiments, appropriate measurements are required.

Szeliski [2010] describes the performance evaluation of a matching strategy by counting the following quantities [see Fawcett, 2006]:

- TP: true positives - defines the number of correct matches. That means, the true corresponding shadow was found.
- FN: false negatives - defines the number of non-matches not correctly rejected. That are shadows which have a corresponding shadow, but no match was proposed, or the proposed match was falsely rejected.
- FP: false positives - defines the number of matches that are incorrect. That are shadow matches, which were assigned to a shadow they do not correspond.
- TN: true negatives - defines the number of non-matches correctly rejected. That are shadows, for which no corresponding shadow exists. Thus no correct match can be found[see Szeliski, 2010].

From this quantities the unit rates True Positive Rate (TPR), False Positive Rate (FPR), Predictive Positive Value (PPV) and Accuracy (ACC) are derived with [Szeliski, 2010]:

$$TPR = \frac{TP}{TP + FN} \tag{30}$$

$$FPR = \frac{FP}{FP + TN} \tag{31}$$

$$PPV = \frac{TP}{TP + FP} \tag{32}$$

[14]The ATON project does not declare any accuracy requirement for the time, but a delay of more than 50 minutes is not expected.

$$ACC = \frac{TP + TN}{P + N} \tag{33}$$

The accuracy ACC is equal to the proportion of the correct matches of the total number of matches. The precision of the matching (PPV) is defined as the ratio between the number of true positives (TP) and the total number of proposed matches (TP + FP).

Szeliski [2010] describes the representation of these quantities and unit rates in a confusion matrix, e.g. presented by figure 31b on page 52. The confusion matrix shows the number of TPs, FPs, TNs and FNs. The first column, containing the number of TPs and FNs, sums up to the number of Positives (P). The number of Negatives (N) is the sum of FPs and TNs. The first row sums up to the number of predicted matches (P') and the second row to the number of predicted non-matches (N')[see Szeliski, 2010].

The performance of the pose estimation can be measured by the offset of the estimated position from the ground truth. The error in each coordinate x,y,z is considered.

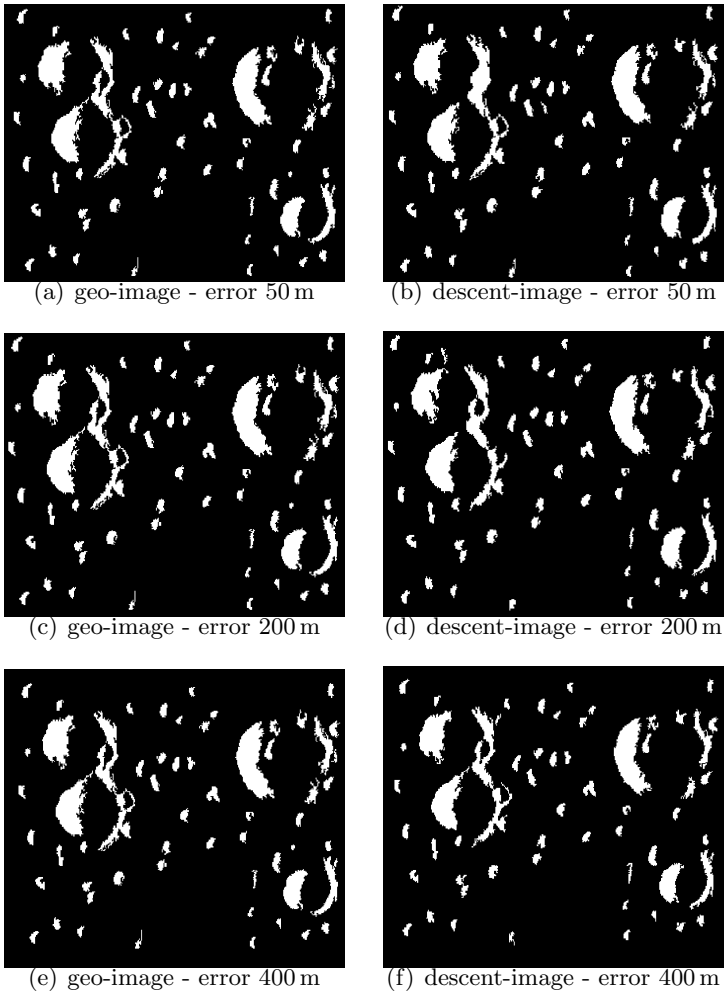
5 Results and analysis

For the three experiments (see section 4.4) the results for the three landing sites (see section4.2) are presented and analysed. Section 5.6 summarizes and discusses the overall results of the BSM.

5.1 Results experiment 1: Robustness with respect to an initial trajectory offset

Experiment 1 (see section 4.4.1) examines the robustness against an initial offset of the planned trajectory. The performance of the approach at offset errors of 50 m, 100 m, 200 m, 300 m, 400 m is evaluated. For details on the experimental set up see section 4.4.1.

Figure 26 shows the binary images for the geo-images and descent-image at different initial trajectory offsets. The larger the offset error, the more the shadows differ in their shape and size. For an offset of 50 m the shadows differ only marginally.



(a) geo-image - error 50 m (b) descent-image - error 50 m

(c) geo-image - error 200 m (d) descent-image - error 200 m

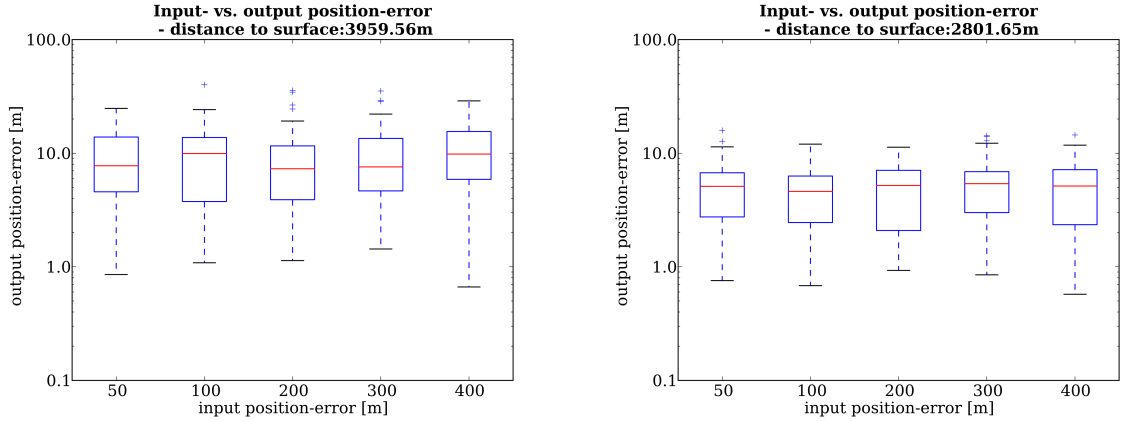
(e) geo-image - error 400 m (f) descent-image - error 400 m

Figure 26: Binary shadow image from different view points: The left images illustrate the binary shadow geo-image, the right images the binary shadow descent-images. With increasing offset of the trajectory, that results in a difference of the view point, the shape of the shadows change.

Landing site model 1

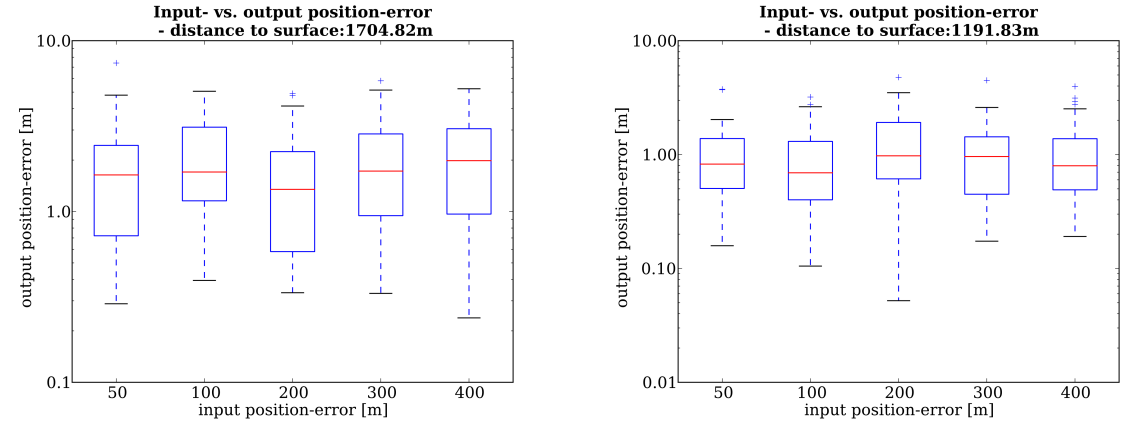
The experiment was repeated 50 times for a section of the trajectory (4 km to 1 km distance to surface) at landing site 1. The landing site is described in detail in section 4.2.1.

Figure 27 shows the error of the estimated position with respect to an initial offset of 50 m, 100 m, 200 m, 300 m, 400 m of the planned trajectory in a box plot[15], for four locations on the trajectory. The y-axis plots the error of the estimated position, and is scaled with log10. The x-axis plots the initial offset from the planned trajectory.



(a) Distance to surface 3959.56 m

(b) Distance to surface 2801.65 m



(c) Distance to surface 1704.82 m

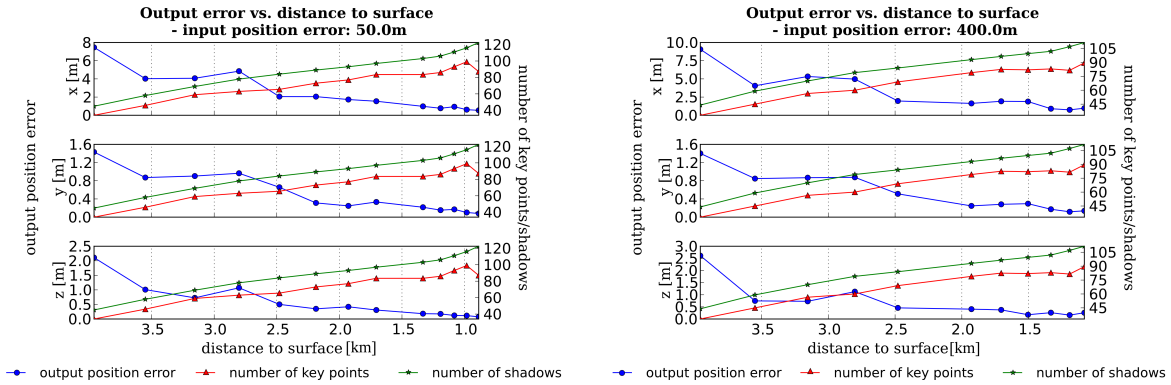
(d) Distance to surface 1191.83 m

Figure 27: Error vs. initial offset of the trajectory offset - landing site 1: Box plots showing the error of the estimated position with respect to the initial offset of 50 m, 100 m, 200 m, 300 m and 400 m for a certain distance to the surface.

Figure 27 implies, that the error of the estimated position gets smaller the closer the surface gets. The median error at a distance of approximately 4 km to the surface is in a range of 8 - 10 m (figure 27a). At a distance of approximately 1 km to the surface the median is smaller than 1 m (figure 27b). Coincidently the deviation of the error gets smaller. The Inter Quartile Range(IQR) - i.e. the difference between the first Quartile and third Quartile - for an initial trajectory offset of 50 m at a distance of approximately 4 km is

[15]The structure of the used box plots and a description of the sample statistics can be found in appendix E.

11 m, at a distance of 1 km the IQR decreased to approximately 1 m.
Further the box plots show that the median has only small fluctuations with respect to the initial offset of the trajectory. No correlation between the accuracy of the estimated position and the initial offset of the trajectory is indicated.
Summarizing figure 27 implies, that the pose estimation gets more accurate and precise the closer the spacecraft gets to the surface. Further no relation between the error of the estimated position and the initial offset of the trajectory is indicated. Thus the method also works well for large changes of the view point position.
The described box plots show only a selection of trajectory locations. Additional box plots for other locations on the trajectory can be found in appendix F.1. They are of the same type and show similar characteristics as the plots described here.



(a) initial offset 50 m (b) initial offset 400 m

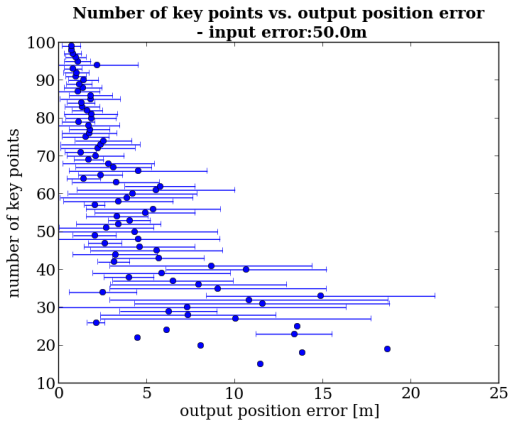
Figure 28: Trajectory error landing site 1 - experiment 1: Plot of the error of the estimated position, number of shadows and number of key points with respect to the distance to the surface. The left y-axis relates to the output position error (blue) and the right y-axis relates to the number of shadows (green) and the number of key points used to estimate the position (red).

Figure 28 shows the error on the x, y, z coordinate of the estimated position, as well as the number of key points and the number of shadows, with respect to the distance to the surface for an initial trajectory offset of 50 m (figure 28a) and 400 m (figure 28b).
The left y-axis plots the output position error on the x, y, z coordinate (blue). The right y-axis plots the number of key points (red) and the number of shadows (green). The number of shadows represents the total number of shadows in the geo-image. The number of key points is equal to the shadows correctly matched. They were used to estimate the position.

The plots show the same characteristics. Similar to the observations on figure 27, the plot implies, that the error gets smaller, the closer the spacecraft gets to the surface. The error magnitude is largest in the x-coordinate. It ranges between 1 m and 7.5 m. Whereas the error on the y-coordinate ranges between 1.5 m and 0.1 m and the error on the z-coordinate between 2.2 m and 0.1 m. Landing site 1 and 2 are both located close to the equator, thus the x-axis is approximately equal to the height over the surface. As the measurement of a laser altimeter is available, the error on the x-coordinate is not of interest.

Further the plots show, that the number of shadows and naturally also the number of key points increases with decreasing distance to the surface. Considering the number

of key points, used to estimate the position, a correlation between the key points and the position error can be observed. With an increasing number of key points the output position error gets smaller. This indicates, that the output position error depends strongly on the number of key points, which depends in this case on the distance to the surface.
Appendix F.1 contains similar plots for the initial offset of the trajectory of 100 m, 200 m and 300 m. They follow the trends stated above.



(a) initial offset of 50 m

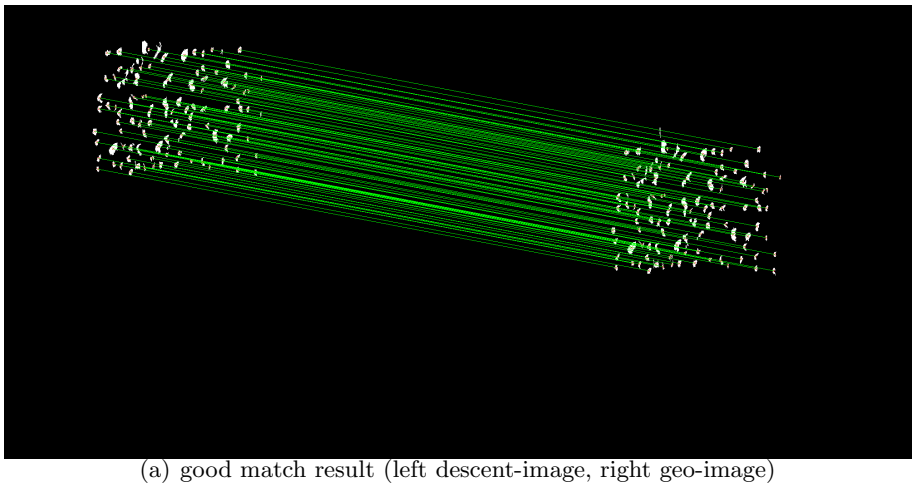
Figure 29: Number of key points vs. error estimated position for landing site 1: Plot of the number of key points with respect to the error of the estimated position. The blue points mark the mean value of the output error and the whiskers show the standard deviation $\pm\sigma$. Points with no whiskers imply, that only one pose estimation was done with this amount of key points.

Figure 29 shows the relationship between the number of key points and the output position error in detail. The y-axis represents the number of key points. The x-axis plots the error of the estimated position. The points show the mean output position error and the whiskers indicate the standard deviation $+\sigma$ and $-\sigma$.

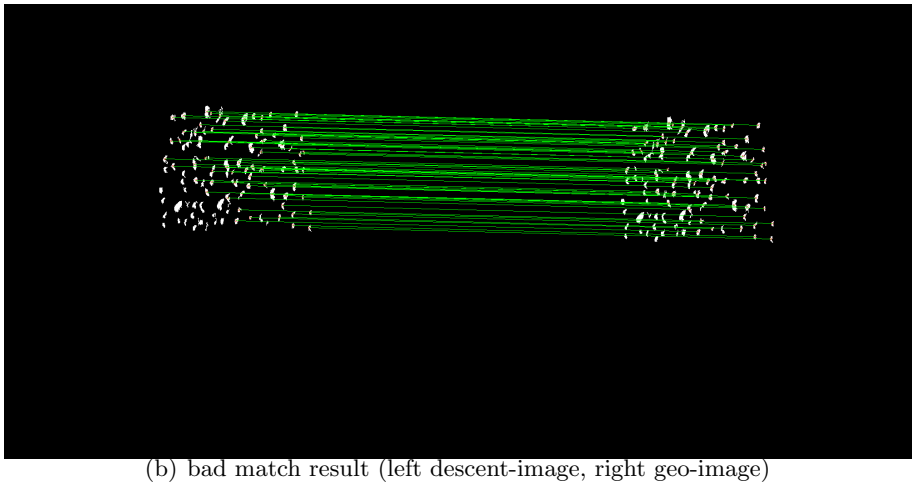
Similar to figure 28, the plot indicates, that a larger number of key points results in a more accurate and more precise position estimate. The estimation with 50 key points results in a mean error of $m = 4\,\text{m}$ with a standard deviation of $\sigma \approx 5\,\text{m}$.
The mean value and standard deviation for the estimation with 70 key points is $m \approx 2\,\text{m}$ and $\sigma \approx 1.5\,\text{m}$. With 100 key points a mean error of $m \approx 1\,\text{m}$ with a standard deviation of $\sigma \approx 0.5\,\text{m}$ was achieved.
Similar plots for an initial offset of the trajectory of 100 m, 200 m, 300 m, 400 m can be found in appendix F.1. They show the same characteristics.

It was shown, that the performance of the pose estimation strongly depends on the number of key points available to estimate the position. For landing site 1 approximately 70 key points are necessary for a precise ($\sigma \approx 1.5\,\text{m}$) and accurate (error $\approx 2\,\text{m}$) position estimate. Depending on the terrain a certain number of shadows exists. The success of the matching decides how many of these shadows can be used as key points for the position estimate. For each execution of the method an image presenting the matches is stored. The lines connect the matched shadow pairs. Figure 30a shows a good example of a final set of matches, after the outlier removal. No false matches were detected. Figure 30b shows a

worse example of a final set of matches, after the outlier removal. A lot of shadows were not matched, e.g. declared as outlier by the RANSAC.



(a) good match result (left descent-image, right geo-image)

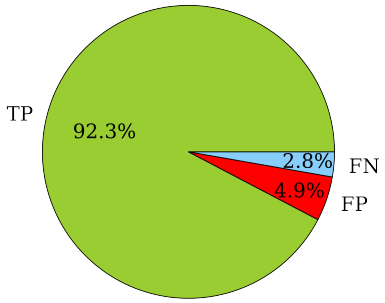


(b) bad match result (left descent-image, right geo-image)

Figure 30: Matching result at landing site 1 -experiment 1: Example of a good a) and bad b) matching result for landing site 1. The lines connect the matched shadow pairs.

To measure the matching performance the confusion matrix was build over all iterations of the experiment. Figure 31b presents the confusion matrix for an initial offset of the trajectory of 50 m before the outlier removal. The corresponding pie chart, figure 31a, shows the percentage of TP, FP, TN and FN.
More than 90% of the matches are TPs. Only 4.9 % are FPs. The TPR rate is close to one. As no TNs exits, the FPR is 1. The PPV has a value of approximately 0.95 and the ACC is about 0.92.
Analyzing the confusion matrices for the trajectory offsets of 100 m, 200 m, 300 m and 400 m (see appendix F.1), a small growth in the FP rate can be detected. For an initial trajectory offset of 400 m 5.6% of the proposed matches are incorrect (FPs).
It was shown, that matching at landing site 1 works robustly, for even large view point changes (up to 400 m). The TPR rate is close to one and the accuracy (ACC) and precision (given by the PPV) of the matching is high.

To remove the false matches, the outliers are detected applying RANSAC. The matches are divided in inliers and outliers. Inliers consist of TPs and not detected FPs. Outliers consist of FPs, FNs and TNs. The outlier removal declares 77.6% of the matches as inliers. 99.1% of these are true matches(TP) and 0.9% are false matches.
The results show, that the outlier removal detects nearly all false matches. A disadvantage is the declaration of a large number of correct matches (TP) as outliers. The RANSAC parameters should be tuned in future.



(a) Percentage of TP, FP, TN, FN

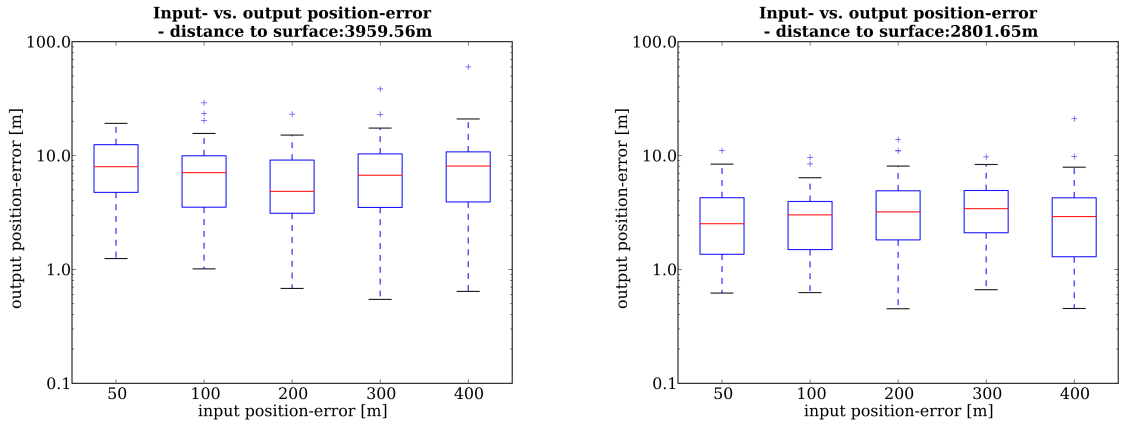
	True matches	True non-matches		
Predicted matches	TP= 53339	FP= 2837	P`= 56176	PVV= 0.95
Predicted non-matches	FN= 1590	TN= 0	N` = 1590	
	P = 54929	N =2287	Total = 57766	
	TPR = 0.97	FPR = 1.0		ACC = 0.92

(b) Confusion matrix

Figure 31: Confusion matrix - trajectory offset of 50 m - landing site 1 - experiment 1 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations

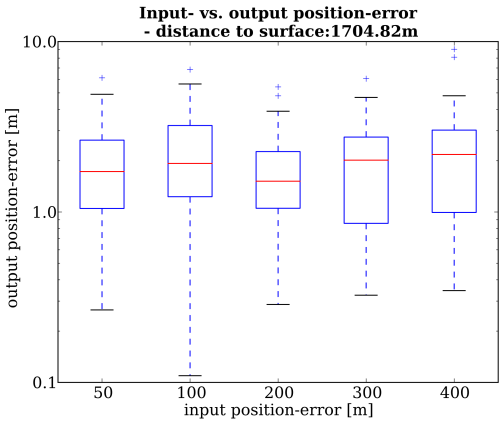
Landing site model 2

Figure 32 shows box plots of the error of the estimated position with respect to initial trajectory offsets of 50 m, 100 m, 200 m, 300 m, 400 m of the trajectory (similar to figure 27).

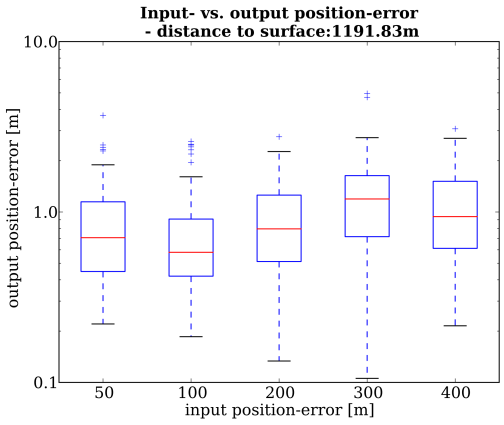


(a) Distance to surface 3959.56 m

(b) Distance to surface 2801.65 m



(c) Distance to surface 1704.82 m



(d) Distance to surface 1191.83 m

Figure 32: Error vs. initial offset of the trajectory offset - landing site 2: Box plots showing the error of the estimated position with respect to the initial offset of 50 m, 100 m, 200 m, 300 m and 400 m for a certain distance to the surface.

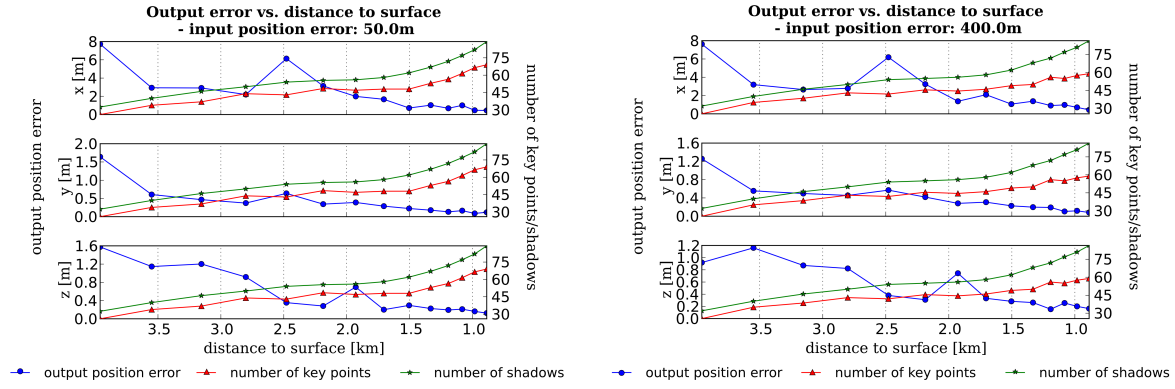
The plots show similar characteristics as the plots for landing site 1 (figure 27). The median error at a distance to the surface of approximately 4 km is about 9 m (figure 32a) and at a distance to the surface of approximately 1 km even smaller than 1 m (figure 32b). The IQR for an initial trajectory offset of 50 m at a distance of approximately 4 km is 11.7 m and at a distance of 1 km 0.85 m.

The results confirm the observations for landing site 1. There seems to be no relation between the initial offset of the trajectory and the error of the estimated position. Further the output position error decreases with decreasing distance to the surface.

The described box plots are only a selection of trajectory locations. Additional box plots for other locations on the trajectory can be found in the appendix F.2. They are of the

same type and show similar characteristics has the plots described.

Figure 33 shows the error on the x, y, z coordinate of the estimated position, as well as the number of key points and the number of shadows, with respect to the distance to the surface for an initial trajectory offset of 50 m (figure 33a) and 400 m (figure 33b).



(a) initial offset 50 m

(b) initial offset 50 m

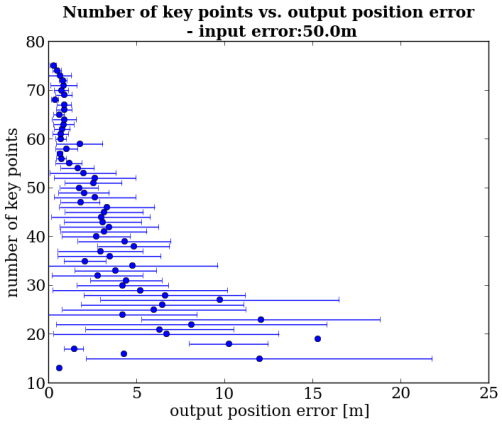
Figure 33: Trajectory error landing site 2 - experiment 1: Plot of the error of the estimated position, number of shadows and number of key points with respect to the distance to the surface. The left y-axis relates to the output position error (blue) and the right y-axis relates to the number of shadows (green) and the number of key points used to estimate the position (red).

The plots show the same characteristics as the plots presented in figure 28. The error gets smaller, the closer the spacecraft gets to the surface. Further the error magnitude is largest in the x-coordinate.
Additionally the number of shadows and respectively the number of key points increases with a decreasing distance to the surface. Similar to the results for landing site 1, the position error seems to decrease with an increasing number of key points. At a distance to the surface of approximately 2.5 km the error in the x coordinate is for both initial errors about 7 m. Currently no explanation can be given for this phenomena.
Appendix F.2 shows similar plots for the initial offset of the trajectory of 100 m, 200 m and 300 m. They follow the general trends stated above.

Similar to figure 29, figure 34 indicates that the accuracy and precision of the pose estimation depends on the number of key points used to estimate the position. Using 50 key points for the estimation resulted in a mean error of $m \approx 2.2\,m$ with a standard deviation of $\sigma \approx 1.5\,m$. With 70 key points a mean error of $m \approx 1\,m$ with a standard deviation of $\sigma \approx 0.5\,m$ was achieved.

Further plots showing the relationship between the number of key points and the error of the estimated position for initial offsets of 100 m, 200 m, 300 m are shown in appendix F.2. Their appearance is similar to the ones in figure 34.

Figure 35 shows an example of a good matching result after the outlier removal. The final set of matches contains only TPs and no FPs can be observed. The highlighted region is an example that shows, where a shadow is missing in the neighbourhood of the descent



(a) initial offset of 50 m

Figure 34: Number of key points vs. error estimated position for landing site 2: Plot of the number of key points with respect to the error of the estimated position. The blue points mark the mean value of the output error and the whiskers show the standard deviation $\pm\sigma$. Points with no whiskers imply, that only one pose estimation was done with this amount of key points.

image. This changes the appearance of the shadow descriptor. Nevertheless all shadows in this region were matched correctly.
Figure 36 shows an example of a bad matching result after the outlier removal. A lot of matches were declared as outliers by the RANSAC.

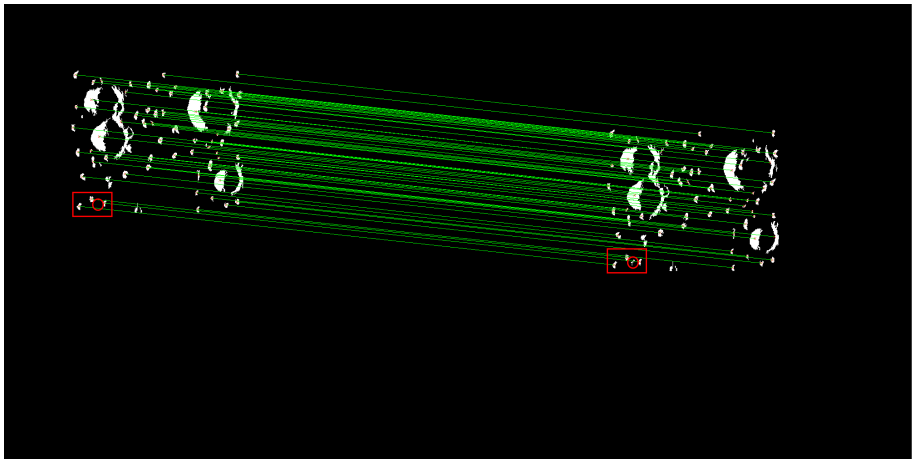


Figure 35: Matching result at landing site 2 - experiment 1: Example of a good matching result for landing site 2. The lines connect the matched shadow pairs. The highlighted regions contains a shadow correctly matched although a shadow is missing in its neighbourhood.

The confusion matrix and the corresponding pie chart, figure 37a and b, show that more than 90% of the matches were TPs and 5.4% are FPs. This leads to a TPR rate close to 1, a FPR rate equal to 1 and a PPV of 0.94 and ACC of 0.93. The high TPR, PPV and ACC indicate a good matching. The outlier removal declared 78.2 % inlier's and 21.8% outliers. The inlier's consist of 99.3% of TPs and 0.7% FPs. Similar to the results of landing site 1, the outlier removal detected nearly all FPs, but also declared about 20% TPs as outliers. For the other initial offsets of the trajectory the confusion matrix can be

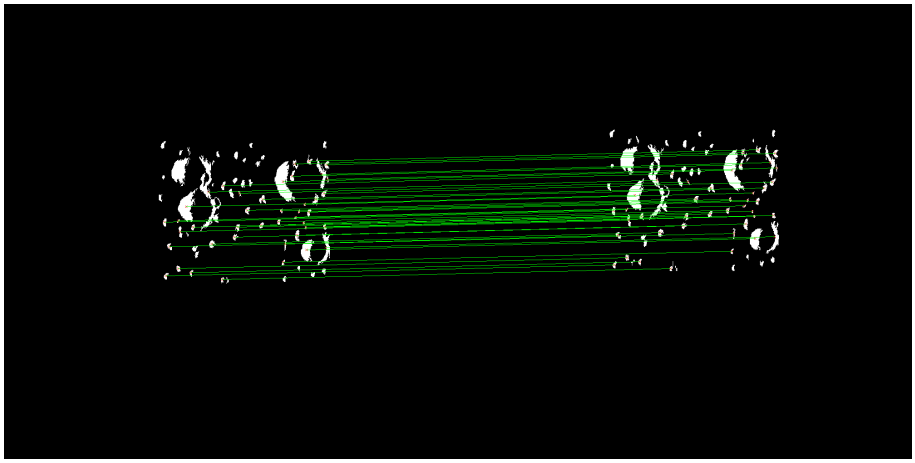
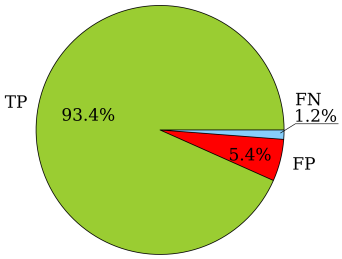


Figure 36: Matching result at landing site 2 - experiment 1: Example of a bad matching result for landing site 2. The lines connect the matched shadow pairs. The highlighted regions contains a shadow correctly matched although a shadow is missing in its neighbourhood.

found in appendix F.2.

It was shown, that the matching for landing site 2 leads to as good results as for landing site 1. The large shadows as well as the small shadows were correctly matched. Shadows, in whose neighbourhood a shadow is missing in one image, could be matched correctly. The outlier removal successfully detected most of the false matches. A disadvantage is, that about 20% of the true matches were also declared as outliers.



(a) Percentage of TP, FP, TN, FN

	True matches	True non-matches		
Predicted matches	TP= 39146	FP= 2283	P`= 41429	PVV= 0.94
Predicted non-matches	FN= 505	TN= 0	N` = 505	
	P = 39651	N = 2283	Total = 41934	
	TPR = 0.99	FPR = 1.0		ACC = 0.93

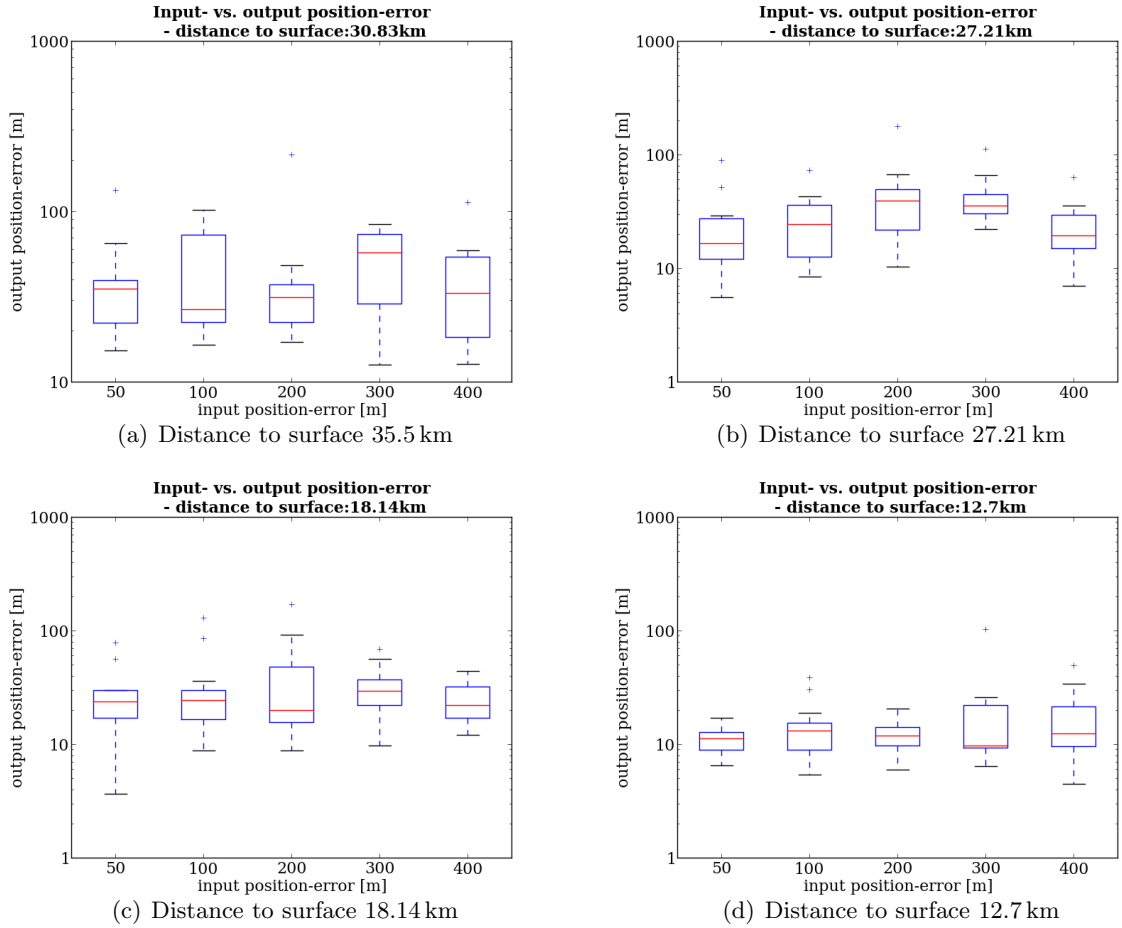
(b) Confusion matrix

Figure 37: Confusion matrix - trajectory offset of 50 m - landing site 1 - experiment 1 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations.

Landing site model 3

Figure 38 shows a significantly larger error of the estimated position, than for landing site 1 and 2. The median for an initial offset of the trajectory of 50 m at a distance to the surface of approximately 30.83 km is approximately 38 m. For a distance of 12.7 km the median is 12 m. This indicates, that the error gets smaller, the closer the spacecraft gets to the surface. The median differs with respect to the initial offset of the trajectory. But the behaviour seems to be random for figure 38a and b.

The described box plots are only a selection of trajectory locations. Additional box plots for other locations on the trajectory can be found in the appendix F.3. They are of the same type and show similar characteristics as the plots described have.



(a) Distance to surface 35.5 km

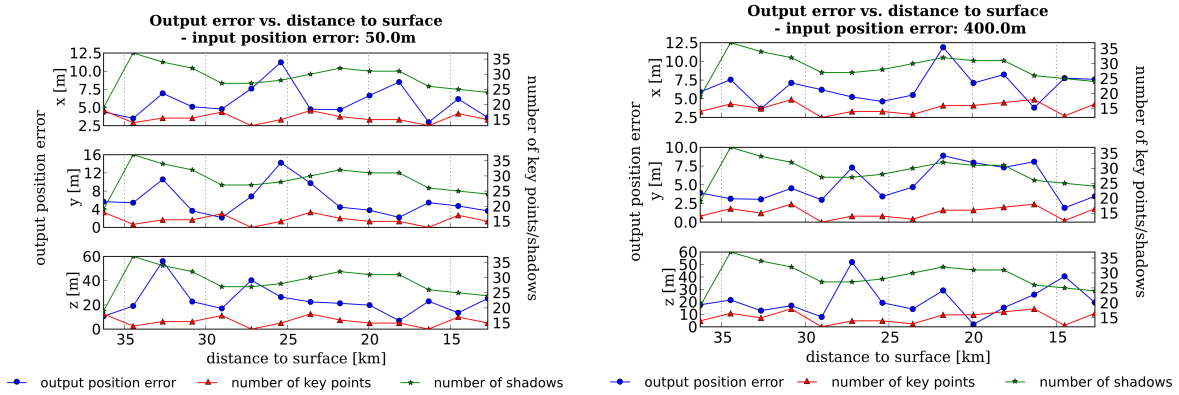
(b) Distance to surface 27.21 km

(c) Distance to surface 18.14 km

(d) Distance to surface 12.7 km

Figure 38: Error vs. initial offset of the trajectory offset - landing site 3: Box plots showing the error of the estimated position with respect to the initial offset of 50 m, 100 m, 200 m, 300 m and 400 m for a certain distance to the surface.

Figure 39 does not confirm a relation between the height and the error of the estimated position. The number of shadows and key points does not increase with a decreasing distance to the surface. The maximum number of shadows is 35. This is about 4 times less than for landing site 1 and 2. Different from the results of landing site 1 and 2, no explicit relation between the amount of key points and the estimated error can be observed. This

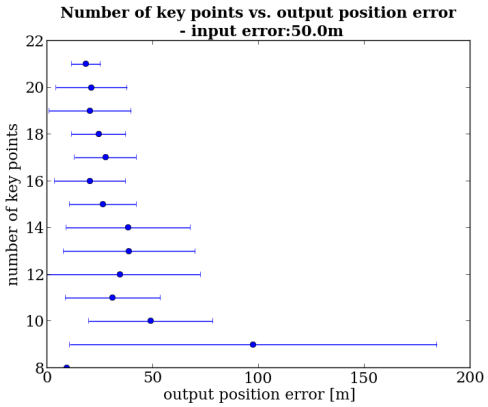


(a) initial offset 50 m (b) initial offset 50 m

Figure 39: Trajectory error landing site 3 - experiment 1: Plot of the error of the estimated position, number of shadows and number of key points with respect to the distance to the surface. The left y-axis relates to the output position error (blue) and the right y-axis relates to the amount of shadows (green) and the number of key points used to estimate the position (red).

is likely because of the smaller amount of key points available. The results of landing site 1 and 2 showed that few key points lead to less accurate and precise results. The same effect is visible here, as the pose estimation is not robust with the available amount of key points detected at landing site 3.

The magnitude of the error for the z-coordinate is largest, instead of for the x-coordinate as for the other landing sites. Landing site 3 is located at latitude $\phi \approx 46°$ and longitude $\lambda \approx 8°$. Thus, different from landing site 1 and 2 the x-coordinate does not correspond to the altitude, but the height over the surface lies in the xz plane. Similar plots for the offset errors of 100 m, 200 m, 300 m are presented in the appendix F.3. They confirm the reported observations.



(a) initial offset of 50 m

Figure 40: Number of key points vs. error estimated position for landing site 3: Plot of the number of key points with respect to the error of the estimated position. The blue points mark the mean value of the output error and the whiskers show the standard deviation $\pm\sigma$. Points with no whiskers imply, that only one pose estimation was done with this amount of key points.

Figure 40 shows the relation between the output error and the number of key points. The maximal number of key points is 21. With 21 points the mean error is approximately $m \approx 20\,m$ with a standard deviation of $\sigma \approx 9\,m$. With less than 10 key points the result is, with a mean error of $m \approx 95\,m$ and a standard deviation of $\sigma \approx 80\,m$, inaccurate and imprecise as it is even more than the initial offset. The plot confirms the observation made in the results of landing site 1 and 2, that the pose estimation gets more accurate and precise the more key points are available.

Figure 41 shows an example of a good matching result at landing site 3. Most shadows were matched correctly. Even some shadows at the edge of the landing site were matched correctly. Other than figure 41, figure 42 shows an example of a worse matching result. Several shadows were matched with the same shadow in the descent image. These shadows all have no or only a few shadows in their close neighbourhood. This might lead to ambiguous descriptors.

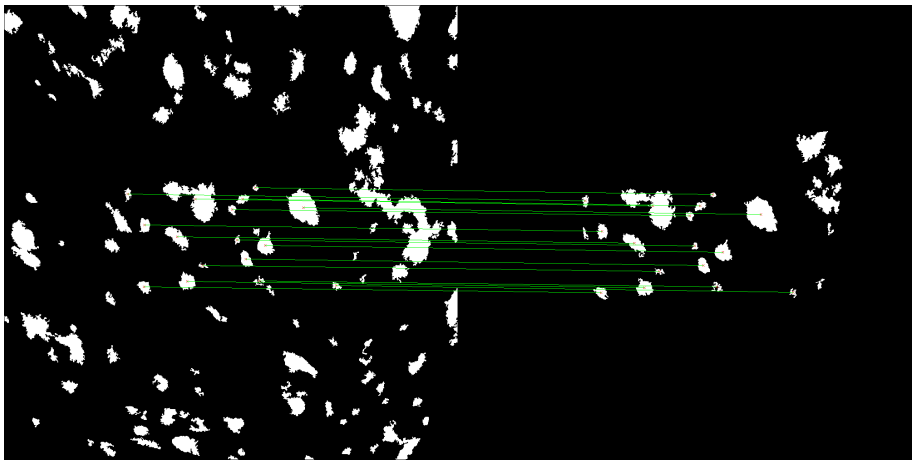


Figure 41: Good matching result for landing site 3 - experiment 1: The lines connect the matched shadow pairs (left descent-image, right geo-image).

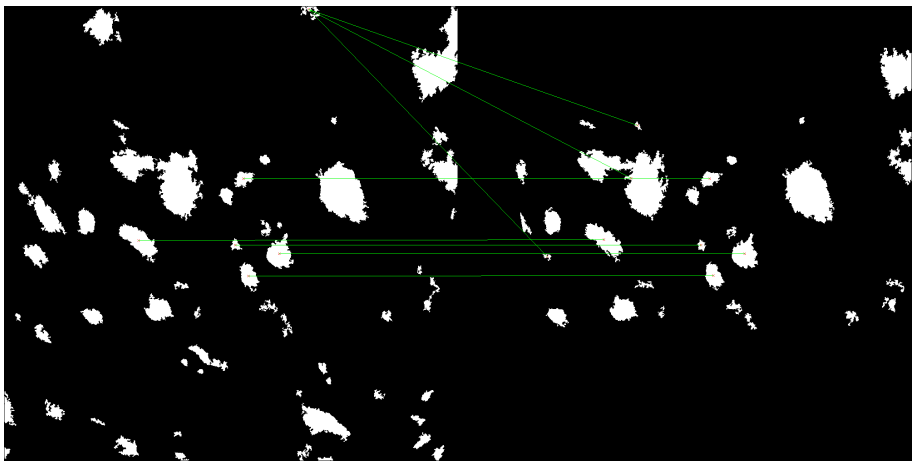
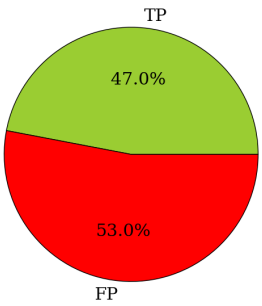


Figure 42: Bad matching result for landing site 3 - experiment 1: The lines connect the matched shadow pairs (left descent-image, right geo-image).

The confusion matrix and the corresponding pie chart (figure 43) shows, that 47% of the

matches were correct (TP) and 53% were false matches (FP). This leads to a precision and accuracy equal to 0.47. Because no FNs and TNs were declared the TPR is equal to 1. The results show that the matching at landing site 3 is not sufficient. This bad performance has mainly three reasons. First, the descent image contains much more content than the geo-image (see figure 25 for illustration). Hence, the neighbourhood of a shadow at the edge of the landing site in the geo-image differs strongly from the neighbourhood of the corresponding shadow in the descent-image. To solve this, the weighted binary comparison was proposed (see section 3.4). This improved the result, as some shadows on the edge of the landing site could be matched correctly (see figure 41). But still a lot of shadows close to the image border could not be matched correctly. Challenging are shadows which are cut of at the image border. Their centroids differ strongly between the geo-image and the descent-image.
Second the matching is based on a nearest neighbour search. Each shadow is matched to the most similar shadow found. A threshold to reject potential mismatches was set. But this seems to be inappropriate.
Third some of the shadows in the image of landing site 3 have no neighbours in their close neighbourhood. The descriptive binary pattern contains no neighbours, thus it is not informative. All shadows with such a neighbourhood were matched to the first shadow with the same pattern, as illustrated in figure 42. A larger neighbourhood could solve this problem. Due to the extensive rendering time of the DEM of landing site 3, the parameters could not be adjusted, yet.

The outlier removal is able to reduce the false matches in the set of inliers by a factor of four. After the outlier removal, the set of inliers consists of 83% TP.



(a) Percentage of TP, FP, TN, FN

	True matches	True non-matches		
Predicted matches	TP= 2478	FP = 2794	P`= 5272	PVV= 0.47
Predicted non-matches	FN= 0	TN= 0	N` = 0	
	P = 2478	N = 2794	Total = 5272	
	TPR = 1	FPR = 1.0		ACC = 0.47

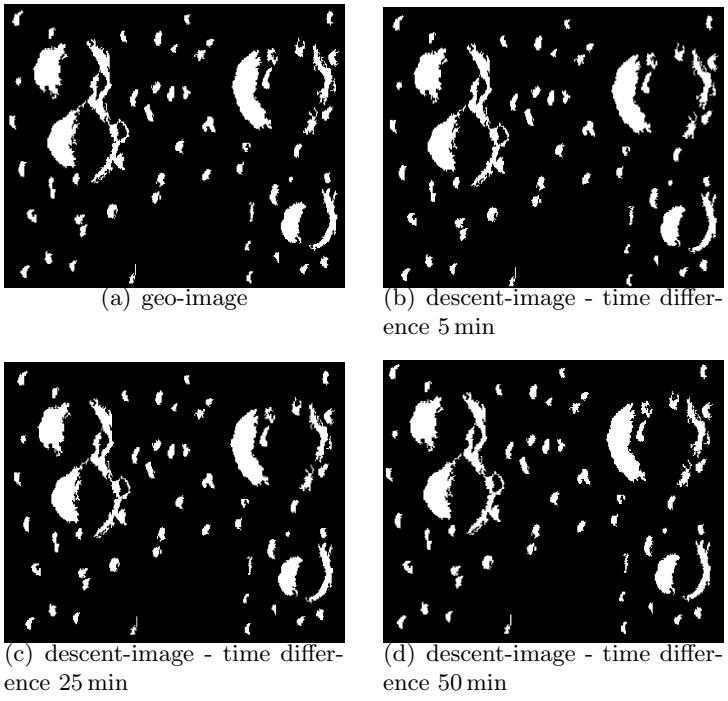
(b) Confusion matrix

Figure 43: Confusion matrix - trajectory offset of 50 m - landing site 3 - experiment 1 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations of experiment 1 with an initial trajectory offset of 50 m.

5.2 Results experiment 2: Robustness with respect to sun angle variations

With experiment 2 the robustness against sun angle variations is evaluated. The geo-image and descent-image were rendered for different times, this caused a change of the sun elevation and azimuth. A time difference of 5, 15, 25, 35, 45 and 50 minutes was tested. It was assumed, that a delay longer than 50 minutes is not realistic for real time missions. This experiment was only run once and not repeated several times, like experiment 1. For details on the experimental set up see section 4.4.2.

Figure 44 shows binary shadow images rendered from equal view points, at different times. It can be seen, that a time difference of 50 minutes only marginally changes the shape of the shadows. The day of the moon is approximately equal to 29,5[16] earth days, thus the sun moves insignificantly within 50 minutes.



(a) geo-image

(b) descent-image - time difference 5 min

(c) descent-image - time difference 25 min

(d) descent-image - time difference 50 min

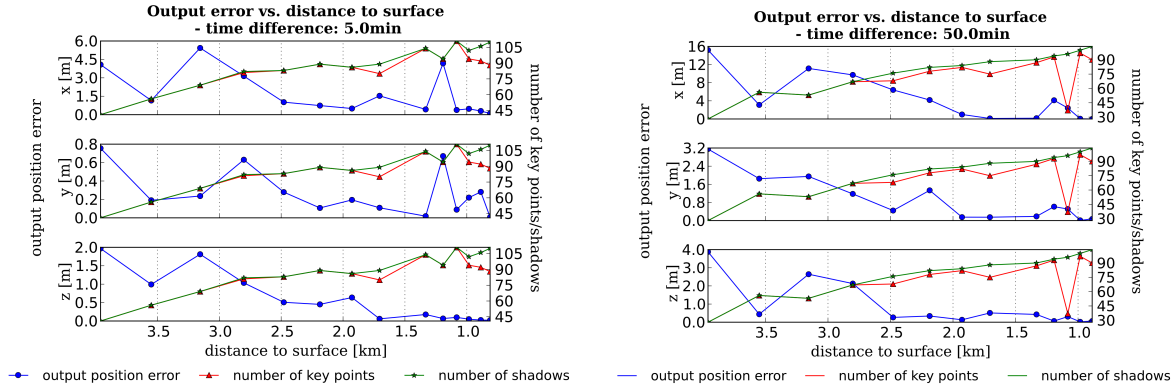
Figure 44: Comparison of binary shadow images with varying sun angles.

Landing site model 1

Figure 45 shows the error on the x, y, z coordinate of the estimated position, as well as the number of key points and the number of shadows, with respect to the distance to the surface for a difference in time of 5 minutes (figure 45a) and 50 minutes (figure 45b). Additional plots for a difference in time of 15, 25, 35, 45 minutes are in appendix F.4

Similar to the results of experiment 1 at landing site 1, the magnitude of the error is largest in the x-coordinate. For a difference in time of 5 minutes the error in the x-coordinate is in range of 5.5 m - 0.5 m, for the y-coordinate 0.8 m-0.05 m and for the z coordinate

[16] synodic day



(a) time difference 5 minutes

(b) time difference 50 minutes

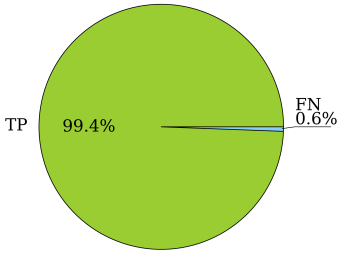
Figure 45: Trajectory error landing site 1 - experiment 2: Plot of the error of the estimated position, number of shadows and number of key points with respect to the distance to the surface. The left y-axis relates to the output position error (blue) and the right y-axis relates to the number of shadows (green) and the number of key points used to estimate the position (red).

2.0 m-0.1 m.
For a difference in time of 50 minutes the error in the x-coordinate is in range 15.8 m-0.8 m, in the y coordinate 3.8 m - 0.2 m and the z- coordinate 3.9 m - 0.2 m. Thus it is about 2.5 times larger on the x- coordinate, 4 times larger on the y-coordinate and 2 times larger on the z-coordinate, as the error at a difference in time of 5 minutes. This indicates a less precise pose estimation with respect to an increasing difference in time.
For a difference in time of 5 min at a distance of approximately 2.8 km and 1.2 km an increase of the error can be observed. But the error is still quite small. The number of key points at this locations is approximately 70 (2.8 km) and 90 (1.2 km). Usually this is an acceptable number of key points.
At a distance to the surface of 1.1 km for a difference in time of 50 minutes the number of key points decreases strongly. Only 40 key points were detected. The outlier removal declared many true matches as outliers.
Further it occurs, that the curve of the number of key points and number of shadows are close to each other. That indicates a "good" matching.

It was shown that the output error of the position estimate increases with increasing difference of time, although the shadows look similar in a visual inspection. At certain locations the error increased, although enough key points were available. No explanation for this phenomena could be found. Overall the pose estimation was sufficiently accurate and precise for the tested differences in time.

The confusion matrices for a difference in time of 5 minutes (figure 46) and 50 minutes (figure 47) confirm the indication of a "good" matching. Before the outlier removal the TP rate for both is close to 100%. No false matches (FP) were proposed for a difference in time of 5 minutes. For 50 minutes only 1.1% FPs were detected. The TPR for 5 minutes is close to 1. As no false positives and true negatives were detected, the FPR is 0. The precision, given by the PPV, is 1 and the accuracy is equal to 0.99. The TPR for a difference in time of 50 minutes is 0.98. The FP rate is about 1%. The PPV is equal to 0.99 and the ACC is 0.97.

The set of inliers, declared by the outlier removal, contains only true matches (TP), all false matches were correctly declared as outliers. But about 20% of the previously correct matches were declared as outliers, too.
The confusion matrices for differences in time of 15, 25, 35 and 45 minutes can be found in appendix F.4.

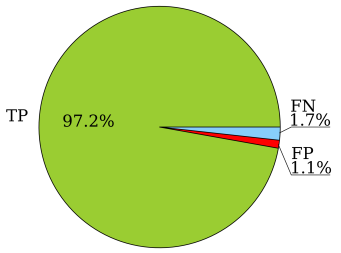


(a) Percentage of TP, FP, TN, FN

	True matches	True non-matches		
Predicted matches	TP= 1185	FP= 0	P`= 1185	PVV= 1.0
Predicted non-matches	FN= 7	TN= 0	N` = 7	
	P = 1192	N =0	Total = 1192	
	TPR = 0.99	FPR = 1.0		ACC = 0.99

(b) Confusion matrix

Figure 46: Confusion matrix - trajectory offset of 5 minutes - landing site 1 - experiment 2 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations



(a) Percentage of TP, FP, TN, FN

	True matches	True non-matches		
Predicted matches	TP= 1003	FP= 11	P` = 1014	PVV= 0.99
Predicted non-matches	FN= 18	TN= 0	N` = 18	
	P = 1021	N = 11	Total = 1032	
	TPR = 0.98	FPR = 1.0		ACC = 0.97

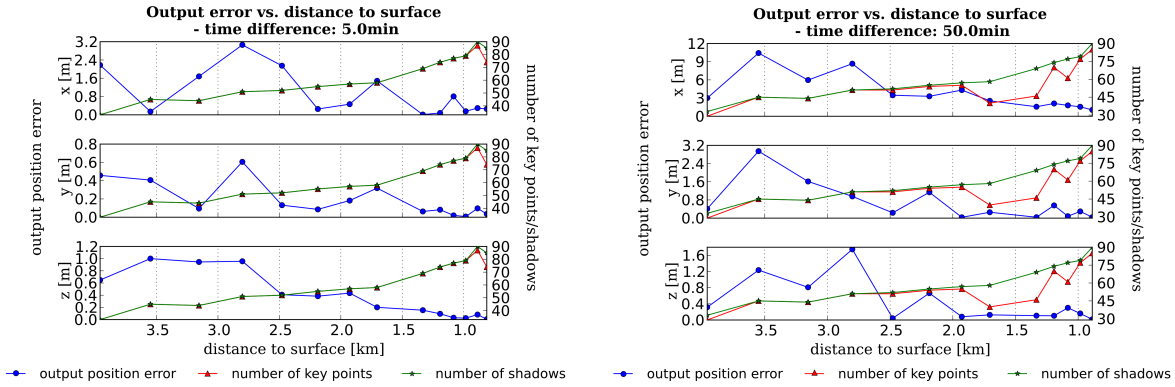
(b) Confusion matrix

Figure 47: Confusion matrix - trajectory offset of 50 minutes - landing site 1 - experiment 2 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations.

Landing site model 2

The results for landing site 2 are similar to the results for landing site 1. Figure 48 shows, that the magnitude of the error is largest in the x-coordinate. For a difference in time of 5 minutes the error in the x-coordinate is in a range of 3.2 m - 0.1 m, for the y-coordinate in range of 0.6 m-0.05 m and for the z-coordinate range of 1.0 m-0.1 m.
For a difference in time of 50 minutes the error in the x-coordinate is in a range of 11.0 m-1.8 m, in the y-coordinate in a range of 3.0 m - 0.2 m and the z-coordinate in a range of 1.8 m - 0.1 m. A relation between the number of key points and the estimated error is not explicitly observable.

The number of shadows and the number of key points is similar, for a time difference of 5 minutes. This indicates that almost all shadows were matched correctly. For a time difference of 50 minutes the number of key points suddenly decreases at a distance to the surface of 1.8 km. An explanation for this sudden decrease of key points could not be found. Similar plots for a difference in time of 15, 25, 35 and 45 minutes can be found in appendix F.5.

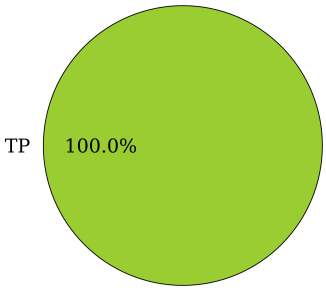


(a) time difference 5 minutes

(b) time difference 50 minutes

Figure 48: Trajectory error landing site 2 - experiment 2: Plot of the error of the estimated position, number of shadows and number of key points with respect to the distance to the surface. The left y-axis relates to the output position error (blue) and the right y-axis relates to the number of shadows (green) and the number of key points used to estimate the position (red).

Figure 49 shows, that the TP rate is 100% for a difference in time of 5 minutes no false matches (FP) were predicted. Thus the TPR, PPV and ACC are equal to 1 and the FPR rate 0. This is the best matching that can be achieved. For a difference in time of 5 min the shadows changed only marginally, the visual inspection of the geo-image and the descent-image showed that both were similar.
Before the outlier removal the TP rate for a difference in time of 50 min is 98.3%, the FP rate is 1.5% and the FN rate is 0.1% (see figure 50). The TPR is 0.99 and the FPR is equal to 1. The precision (PPV) is 0.98 and the accuracy (ACC) is about 0.98. The outlier removal detected the false matches correctly. The set of inliers contained only TPs.

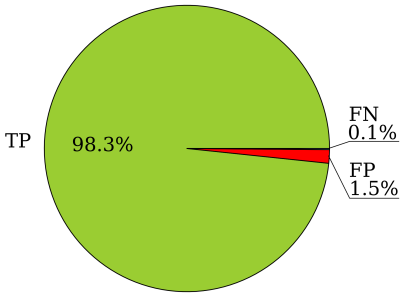


(a) Percentage of TP, FP, TN, FN

	True matches	True non-matches		
Predicted matches	TP= 855	FP= 0	P`= 855	PVV= 1.0
Predicted non-matches	FN= 0	TN= 0	N`= 0	
	P = 855	N =0	Total = 855	
	TPR = 1.0	FPR = 0		ACC = 1.0

(b) Confusion matrix

Figure 49: Confusion matrix - trajectory offset of 5 minutes - landing site 2 - experiment 2 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations



(a) Percentage of TP, FP, TN, FN

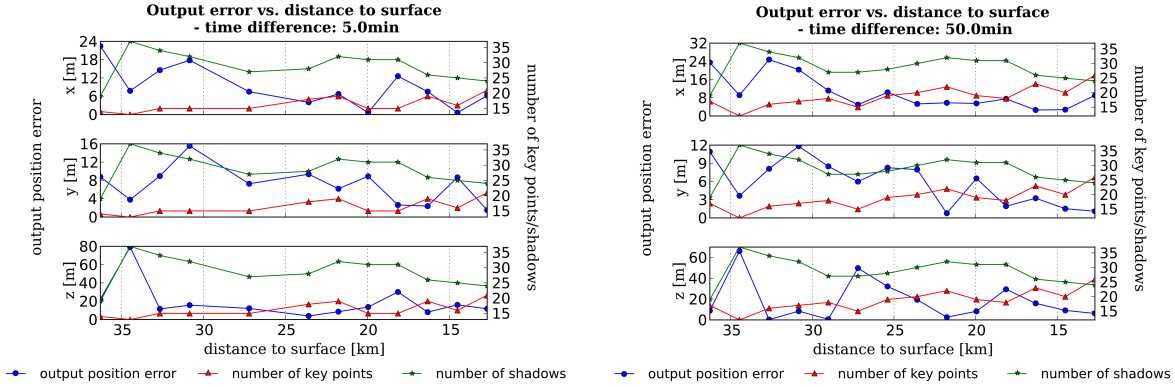
	True matches	True non-matches		
Predicted matches	TP= 774	FP= 12	P`= 786	PVV= 0.98
Predicted non-matches	FN= 1	TN= 0	N`= 1	
	P = 775	N = 12	Total = 787	
	TPR = 0.99	FPR = 1.0		ACC = 0.98

(b) Confusion matrix

Figure 50: Confusion matrix - trajectory offset of 50 minutes - landing site 2 - experiment 2 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations.

Landing site model 3

Figure 51 shows the error on the x, y, z coordinate of the estimated position, the number of key points used to estimate the position and the number of shadows, with respect to the distance to the surface for a difference in time of 5 minutes and 50 minutes.
The magnitude of the error is largest in the z-coordinate, the maximal error is 80 m for a difference in time of 5 minutes and 77 m for a a difference in time of 50 minutes. It is assumed, that the small number of key points is responsible for the unstable results. It can be observed that a peak in the error curve, e.g. at a distance to the surface of 18 km and 16 km for 5 minutes difference in time, comes with less key points.



(a) time difference 5 minutes

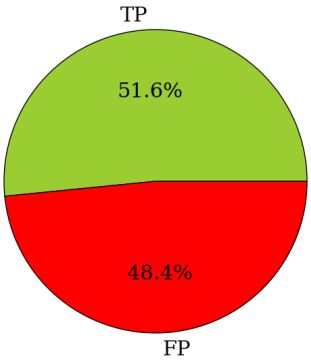
(b) time difference 50 minutes

Figure 51: Trajectory error landing site 3 - experiment 2: Plot of the error of the estimated position, number of shadows and number of key points with respect to the distance to the surface. The left y-axis relates to the output position error (blue) and the right y-axis relates to the number of shadows (green) and the number of key points used to estimate the position (red).

The confusion matrix for the experiment with 5 minutes difference in time, shown by figure 52, shows that 51.6% of the matches were declared as TP. The remaining matches were declared as FPs. This leads to a low precision and accuracy. The PPV and ACC are equal to 0.52.

Figure 53 shows the confusion matrix and pie chart for a difference in time of 50 minutes. 43% FPs were proposed, this is less than for a difference in time of 5 minutes. This was not expected, as the shadows for a difference in time of 50 minutes are assumed to change their shape stronger, which complicates the matching.

Overall the matching at landing site 3 was not sufficient. Often only half of the matches were correct. The accuracy and precision is very low. The same reasons, as for experiment 1 apply (see paragraph 5.1 p. 57).

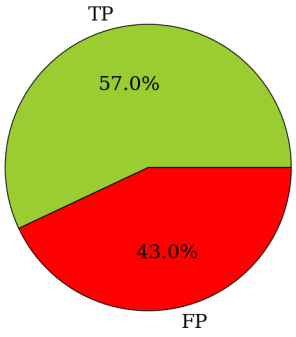


(a) Percentage of TP, FP, TN, FN

	True matches	True non-matches		
Predicted matches	TP= 197	FP = 185	P`= 382	PVV= 0.52
Predicted non-matches	FN= 0	TN= 0	N` = 0	
	P = 197	N = 185	Total = 382	
	TPR = 1.0	FPR = 1.0		ACC = 0.52

(b) Confusion matrix

Figure 52: Confusion matrix - trajectory offset of 5 minutes - landing site 3 - experiment 2 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations.



(a) Percentage of TP, FP, TN, FN

	True matches	True non-matches		
Predicted matches	TP= 261	FP = 197	P`= 458	PVV= 0.57
Predicted non-matches	FN= 0	TN= 0	N` = 0	
	P = 261	N = 197	Total = 458	
	TPR = 1.0	FPR = 1.0		ACC = 0.57

(b) Confusion matrix

Figure 53: Confusion matrix - trajectory offset of 50 minutes - landing site 3 - experiment 2 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations.

5.3 Results experiment 3: Robustness to a trajectory offset and sun angle variations

Experiment 3 combines experiment 1 and experiment 2. The difference in time and the position error are randomly generated. The maximal offset of the trajectory is 100 m and the maximal difference in time is 50 minutes. For details on the experimental set up and parameters see section 4.4.3.

Landing site model 1

Figure 54 is similar to figure 28. The error on the x, y, z coordinate and the number of shadows and key points are the mean values of all trials. It can be observed, that the error on each coordinate of the estimated position decreased with a decreasing distance to the surface. With decreasing distance to the surface, the number of visible shadows increased. About 75% of the available shadows were declared as key points.

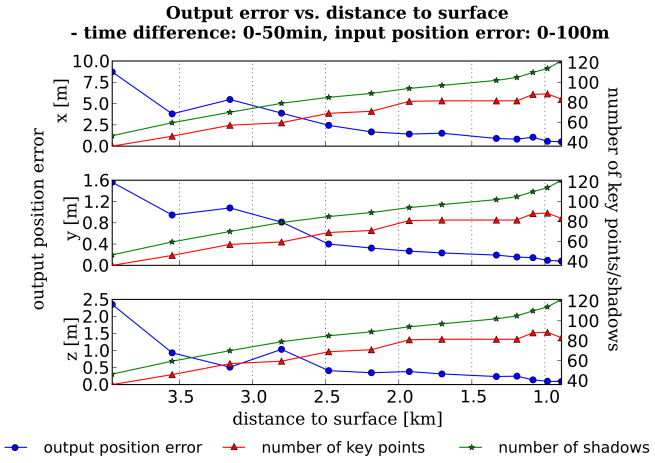
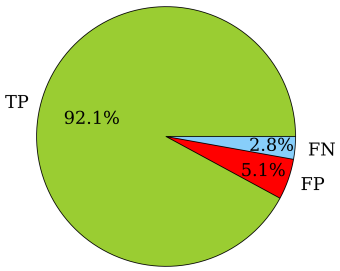


Figure 54: Trajectory error landing site 1 - experiment 3: Plot of the error of the estimated position, number of shadows and number of key points with respect to the distance to the surface. The left y-axis relates to the output position error (blue) and the right y-axis relates to the number of shadows (green) and the number of key points used to estimate the position (red).

Figure 55 shows the confusion matrix for all iterations. The TP rate is 92.1%. 5.1% FPs and 2.8% FNs were detected. The TPR is close to 1 with 0.97. The PPV is 1 and the ACC is 0.92. After the outlier removal, the set of inliers (77.3% of all proposed matches) consisted of 99.1% true matches (TP) and 0.9% false matches (FP).

The results of experiment 3 at landing site 1 show, that a difference in time of up to 50 minutes and an initial offset of the trajectory of up to 100 m can be handled well. About 90% of the shadows were correctly matched.



(a) Percentage of TP, FP, TN, FN

	True matches	True non-matches		
Predicted matches	TP= 95928	FP= 5276	P`= 101204	PVV= 0.95
Predicted non-matches	FN= 2930	TN= 0	N` = 2930	
	P = 98858	N =5276	Total = 104134	
	TPR = 0.97	FPR = 1.0		ACC = 0.92

(b) Confusion matrix

Figure 55: Experiment 3 at landing site 1 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations.

Landing site model 2

Figure 56 shows similar characteristics as figure 54 for landing site 1. The number of shadows and key points increased and the error decreased with a decreasing distance to the surface. At approximately 2.8 km and 2.5 km distance to the surface the error increases against the trend. The reason for this is not clear.

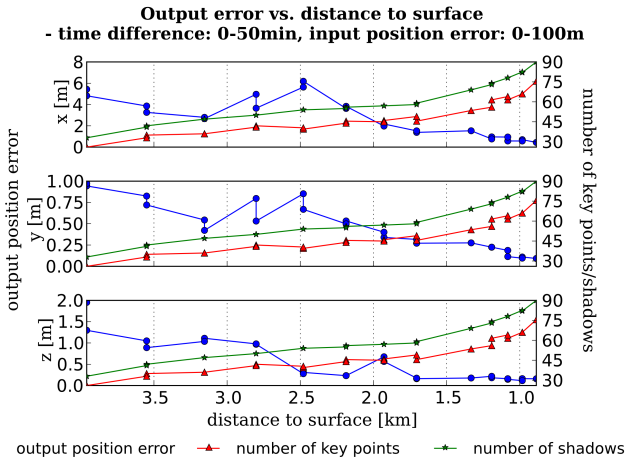
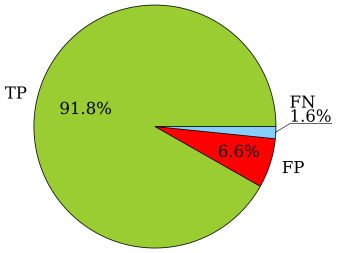


Figure 56: Trajectory error landing site 2 - experiment 3: Error of the estimated position, number of shadows and number of key points with respect to the distance to the surface. The left y-axis relates to the output position error(blue) and the right y-axis relates to the number of shadows(green) and the number of key points used to estimate the position (red).

Figure 57 shows the confusion matrix for all iterations. The TP rate is 91.8%. 6.6% FPs and 1.6% FNs were detected. The TPR is close to 1 with 0.98. The PPV is 0.93 and the

ACC is 0.91. After the outlier removal the FP rate of the inlier's is about 0.7%.

Similar to the results for landing site 1, the results of experiment 3 at landing site 2 show that a difference in time up to 50 minutes and an initial offset of the trajectory up to 100 m can be handled well.



(a) Percentage of TP, FP, TN, FN

	True matches	True non-matches		
Predicted matches	TP= 65287	FP= 4685	P`= 69972	PVV= 0.93
Predicted non-matches	FN= 1141	TN= 0	N` = 1141	
	P = 66428	N =5276	Total = 71704	
	TPR = 0.98	FPR = 1.0		ACC = 0.92

(b) Confusion matrix

Figure 57: Experiment 3 at landing site 2 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations.

Landing site model 3

Similar to the results of experiment 1 and 2 for landing site 3, figure 58 shows no explicit relation between the distance to the surface and the error of the estimated position. A comparison of the number of key points with the output position error indicates a dependency of both. An increase of the error often comes with a decrease of the key points. The number of shadows first increased with respect to the distance to the surface, but decreased once the spacecraft got closer than 16 km.
Again the magnitude of the error on the z-coordinate is largest as it ranged between 40 m and 14 m.

Figure 59 shows that on average half of the matches were TPs and half of them were FPs. Thus, also the precision (PPV) and accuracy (ACC) was equal to 0.5. The outlier removal was able to reduce the percentage of FPs in the set of inlier's on average to 13%.

The results for landing site 3 are not sufficient. Only half of all shadows could be matched correctly. The outlier removal was able to detect most of the false matches. But the maximal number of key points was about 20, which is by far to less for an accurate and precise position estimate.

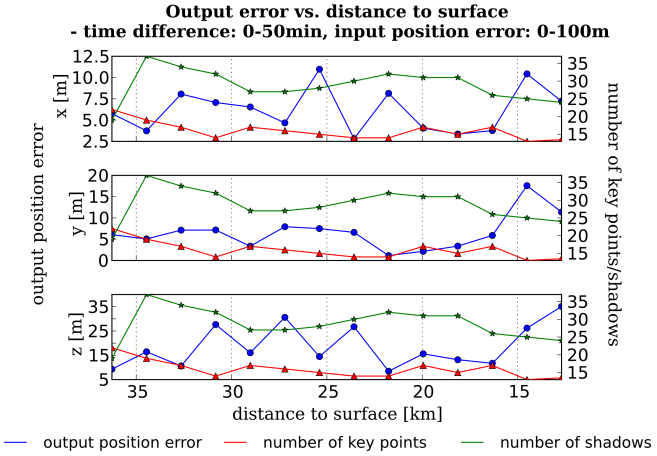
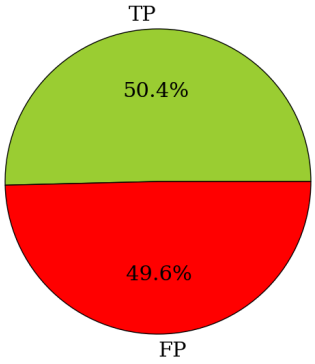


Figure 58: Trajectory error landing site 3 - experiment 3: Plot of the error of the estimated position, number of shadows and number of key points with respect to the distance to the surface. The left y-axis relates to the output position error (blue) and the right y-axis relates to the number of shadows (green) and the number of key points used to estimate the position (red).



(a) Percentage of TP, FP, TN, FN

	True matches	True non-matches		
Predicted matches	TP = 2170	FP = 2137	P' = 4307	PVV = 0.50
Predicted non-matches	FN = 0	TN = 0	N' = 0	
	P = 2170	N = 2137	Total = 4307	
	TPR = 1	FPR = 1		ACC = 0.50

(b) Confusion matrix

Figure 59: Experiment 3 at landing site 3 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations.

5.4 Computation time

The method was tested on an computer with a Intel(R) Xeon(R) CPU W3530 processor, with the OS Suse Linux Enterprise Desktop 11 64bit and 5975MB RAM. Table 7 shows the mean computation time achieved with the used hardware for experiment 1 at landing site 1 at an altitude of 3154.7 m ($\approx$ 70 shadows per image) for the main processes. Most of the time was consumed by the outlier removal. The outlier removal uses RANSAC, which is an iterative approach and naturally slow. It could be speed up by reducing the number of allowed iterations. However, this influences the quality of the result.
For comparison, the time of the shadow matching with and without pyramidal implementation was measured. The pyramidal implementation enhanced the speed, but considering its complexity, the pyramidal matching process should even be faster. This is likely because, the current implementation was not optimized with respect to computational time. Although the pose estimation is an iterative approach, it could be shown, that it converged fast (mean time $\approx$0.18 sec).
The mean computation time for gMET and the shadow extraction and description was fast. This steps can possibly done on board the spacecraft.

Table 7: Computation time for each processing step

Process	**mean time[sec]**
gMet	0.68
Shadow extraction and description	0.0085
Shadow Matching	6.75
Shadow Matching Pyramid	1.49
Pose Estimation	0.18
Outlier removal	3.3
Sum normal implementation	$\sum = 10.91$
Sum pyramidal implementation	$\sum = 5.65$

5.5 Memory consumption

Two possibilities exist for storing the necessary data. Either the binary geo-image can be stored or the already extracted shadow descriptors.
Storing the binary images leads to a fixed memory storage for each image. It only depends on the image size w x h. A binary shadow image of size 1024 px x 1024 px requires 128 kbyte. For the descent from approximately 4 km to approximately 1 km (time $\approx$ 1000 seconds) the total required memory would be 12.5 Mb if the BSM is applied every 10 seconds.
The required memory, for storing the descriptors, depends on the number of shadows n_s, the radius r of the neighbourhood which is encapsulated, the number of pyramid layers $\mathbf{G}_{\kappa=\mathbf{1}\ldots\mathbf{l}}$ and their resolution $g_{x,\kappa}, g_{y,\kappa} = r/spacing_\kappa$. For one image the required memory

can be calculated as follows

$$memory = 1\,\text{bit} \cdot n_s \cdot \sum_{\kappa=0}^{kappa=l} (g_{x,\kappa} \cdot g_{y,\kappa}) \tag{34}$$

For 80 shadows, a neighbourhood radius $r = 200px$ and three pyramid layers $l = 3$ with spacings of 20, 40 and 80 px, results in a descriptor size of 5.13 kbyte.
If a large number of shadows is expected it would be better to store the binary image, as the memory storage is constant. Else the descriptors can be stored.

5.6 Discussion

Goal of the thesis was to find a monocular absolute localization approach, which is applicable until or shortly before touchdown, where the current pose estimation methods of ATON can not be used. Further the approach should be applicable to different types of terrain, require low memory storage and low computation time.

The results of experiment 1 show that the position could be estimated accurately and precisely at landing site 1 and 2. The achieved position accuracy at landing site 3 was not sufficient. The accuracy of the pose estimation strongly depended on the number of key points used to estimate the position. More than 90% of the shadows were matched correctly with the proposed matching strategy. The addition and deletion of a few shadows did not affect the position error significantly. The shadow matching at landing site 3 was more challenging. At the border of the geo-image the shadows had fewer shadows in their neighbourhood than the corresponding shadows in the descent-image. The weighted binary comparison mismatched some of these shadows. Only half of the shadows at landing site 3 could be matched correctly, because many shadows had not enough close neighbours. A larger neighbourhood should be considered in the future to compensate this effect.

The results of experiment 2 show that a time difference of up to 50 min between expected and real position of the spacecraft reshapes the shadows only marginally. The matching and pose estimation were influenced insignificantly. A time difference of 50 min still lead to sufficient results.

The results of experiment 3 at landing site 1 and 2 show that the method can handle sun angle variations of up to 50 min and offsets of the trajectory of up to 100 m. The results for landing site 3 are, similar to the results of experiment 1 and 2, insufficient. The TPR rate of the matching is too low and thus the position estimate less accurate and precise.

It was shown that the proposed BSM is applicable to different terrain. The terrain of landing site 1 and landing site 2 contains only craters. Landing site 1 contains numerous small craters and landing site 2 contains several small and four big craters. Landing site 3 is a hilly region. At all landing sites the shadows could be matched correctly. However the method is restricted to rough terrain, as the pose estimation strongly depends on the number of key points (shadows). But since the goal of future missions is to land close to hazardous and rough terrain, the method is well suited for these missions. Additionally, a more robust method for the pose estimation might enhance the results significantly. Such a method should require only a minimum amount of key points for an accurate position

estimate. Considering the design of the BSM implementation (refer to figure 11 on p. 20), the pose estimation method can easily be exchanged in a future optimization.

A measured total mean computation time of 5.65 seconds with a standard deviation of approximately 1 seconds was achieved. The most time consuming steps are the matching and the outlier removal. The matching speed was enhanced with the pyramidal implementation, resulting in a speed up of ≈ 5.26 seconds . The outlier removal costs computation time due to its iterative approach. Possibly more parameter tuning might increase the speed. For the ATON project currently a minimum computation time lower 10 seconds is required, the current BSM implementation fullfills this. Furthermore, the current implementation was done with Python and is not optimized for real time speed. Hence, a future optimization can lead to a shorter computation time.

6 Summary and Conclusion

This thesis proposes Binary Shadow Matching (BSM), a monocular vision based pose estimation system based on matching binary shadows. The system is designed for the usage during lunar landings. It is intended to enhance the optical navigation system ATON, especially during the final stage of the descent. A prototype of the system was implemented and evaluated in software simulation to serve as a proof of concept. The system requires two images as input, a geo-referenced image of the scene and an image taken by the camera of the spacecraft during the descent. The shadows of both images are extracted, described and matched. Based on the point correspondences from the matching, the pose is estimated. Output of the system is the absolute position and a value for its confidence.

State of the art features, used by optical navigation approaches, e.g. SIFT [Trawny et al., 2007, Johnson et al., 2007] or craters [Cheng and Ansar, 2005], are complex to extract and are not illumination invariant. Moreover, known craters, get sparse, the closer the spacecraft gets to the surface. The shadow features used by the BSM can be extracted very fast and require less memory storage. Further shadows can be seen from any altitude. Because the geo-images, i.e. the reference images, are rendered from a DEM with similar illumination as expected during the landing, the BSM is independent of illumination changes.

The performance and limitations of the implementation were evaluated in three experiments, each done on three different landing sites. Small variations, resulting in addition and deletion of a few shadows, can be compensated. The matching fails for neighbourhoods where a lot of shadows are missing in one of the two images. However the false matches could be removed successfully with the RANSAC outlier removal approach.
In addition, the implementation matches nearest neighbours and marks them as a mismatch, when the similarity is below a threshold. The chosen threshold is not optimal, yet. An appropriate threshold or even an adaptive threshold method should be found to reduce the number of mismatches.

The estimated pose is absolute, accurate and precise, if more than 70 key points are available. The approach is restricted to rough terrain with enough shadows, but it is independent of the type of rough terrain. The method is well suited for future missions whose goal is to land close to hazardous terrain of high scientific value.
It was shown that the approach is invariant to sun angle variations, caused by differences in time of up to 50 minutes. A longer time difference is not expected for real missions. A difference of 50 minutes might occur due to a small delay, a small deviation of the trajectory or minor mission disturbances. Major interferences, which cause a long time shift of the landing are not considered.
The reference data is invariant to illumination differences as it is derived from DEMs. For each mission the images are rendered exclusively for the assumed time, sun elevation and azimuth.
The mean computational time is about 5.65 seconds, which corresponds to the objective time of 10 seconds for the 3D matching method of ATON. Most time is spend removing the outliers with the iterative RANSAC algorithm. The pose estimation is also an iterative approach, which is accurate, but can be slow to converge. It would be interesting to see if the computation time can be decreased by using a non-iterative approach, e.g. the

EPnP [Lepetit et al., 2007], without a significant quality loss. Further the pose estimation is based on least squares, which is sensitive to outliers. A more robust minimization function such as M-estimators could improve the estimation result.

The required memory of the approach is relatively low compared to grey scale images or DEMS, since only binary vectors of the shadow descriptions need to be stored. Moreover the required memory storage for each shadow is equal.

This thesis was a first investigation of shadow based matching for spacecraft localization during lunar landings. It would be interesting to further investigate real lunar shadows. In this thesis the shadows were created with simulated data. It should be investigated how well the simulated shadows can be matched with real shadows. Another interesting point would be to find the final limitations of the BSM. The approach is not restricted to any terrain, one could also imagine to evaluate the performance and limitations of the approach for landings on other planetary bodies. For planets with an atmosphere the shadows could be rendered with a global illumination model.

Future exploration missions require a precise landing, which itself requires accurate and absolute localization until shortly before touchdown. The proposed BSM is a promising approach for this challenge and the results of this work show that it can be a valuable contribution to future vision based planetary landing systems.

A Camera model

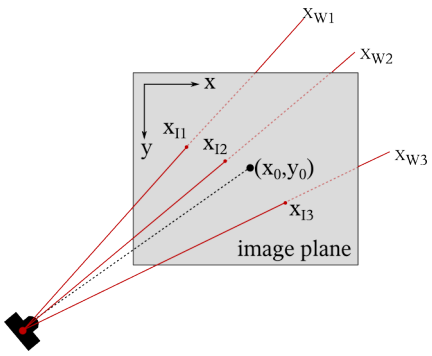


Figure 60: Projection to the image plane: Each 3D world point $x_{W_1}, x_{W_2}, x_{W_3}$ is projected with the projection equation 35 to a 2D image point$x_{I_1}, x_{I_2}, x_{I_3}$. (x_0, y_0) is the principal point of the image. (figure based on [Rodehorst, 2004, figure 4.6 b, p.69])

Let $\mathbf{x_W}$ be $n = 1...N$ homogeneous world points and $\mathbf{x_I}$ the projection of $\mathbf{x_W}$ to the image plane as illustrated in figure 60. Their relationship is expressed by [Hartley and Zisserman, 2003, p.156]

$$\mathbf{x_I} = \mathbf{P}\mathbf{x_W} \text{ with } \mathbf{P} = \mathbf{K}[\mathbf{R}|\mathbf{t}] \tag{35}$$

where $\mathbf{P}$ is the camera or projection matrix. $\mathbf{P}$ can be decomposed in the parameters of the intrinsics $\mathbf{K}$ and extrinsics $\mathbf{R}$, $\mathbf{t}$.

The intrinsic or calibration matrix $\mathbf{K}$ represents the inner orientation of the camera and is of the form [Hartley and Zisserman, 2003]:

$$\begin{bmatrix} \alpha_x & s & x_0 \\ 0 & \alpha_y & y_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{36}$$

where [Hartley and Zisserman, 2003]

- *"α_x is the scale factor in the x-coordinate direction,*
- *α_y is the scale factor in the y-coordinate direction,*
- *s is the skew,*
- *$(x_0, y_0)^T$ are the coordinates of the principal point."*

The extrinsic parameters comprise the orientation $\mathbf{R}$ of the camera and the position $\mathbf{t}$ of the camera center in the world coordinate system. $\mathbf{R}$ is a rotation matrix and $\mathbf{t}$ is a translation vector. Applying the rigid transform

$$\mathbf{x_C} = \mathbf{R}[\mathbf{I}| - \mathbf{t}]\mathbf{x_W} \tag{37}$$

to a point $\mathbf{x_W}$ in the world coordinate system, it is transformed to the camera coordinate system (subscript $_C$). The camera model is explained in detail in the textbook of Hartley and Zisserman [2003].

B Least squares and error theory

In the following the basics of error propagation and least squares are introduced.

Observations and Errors
All observed quantities contain errors, which are classified as either systematic or random [Ghilani, 2011]. The term outlier - in geodesy often denoted as blunder - is often referred to the error context, but does not fall within the error definition. The three terms outlier, systematic error and random error can be defined as follows:

Outliers are mistakes caused by, e.g. misinterpretations of the observer, failure of equipment or failure of an algorithm. They are treated as observations that do not fit to the rest of the observed quantities [Mikhail and Ackermann, 1982]. By checking the consistency of a set of observations, outliers can be treated before the observations are used for further calculations [Mikhail and Ackermann, 1982]. In the context of this thesis outliers are false matches, i.e. proposed matches that are incorrect.

Systematic errors are constant errors with equal sign. They usually follow a physical law [Ghilani, 2011]. For example, wrong calibrated instruments can cause systemic errors [Ghilani, 2011]. In the context of this thesis systematic errors may occur due to a wrong camera model. It is tried to avoid systematic errors by precisely setting up the system.

Random errors are produced by unpredictable and unknown variations in the experimental set up. They differ in magnitude and sign and can be treated statistically [see Ghilani, 2011]. In the context of this thesis random errors are variations in the key point coordinates returned by the matching process.

Functional model
A functional model relates the observations with the unknown and known parameters. Let $\mathbf{l_i}$ be an observation, X be the set of known parameters and U the set of unknown parameters, then the functional relationship is given as

$$\mathbf{l_i} = f_i(X, U) \text{ with } i = 1...n \tag{38}$$

where n is the number of observations. The model follows the Gauss-Markov model, as each equation contains only one observation $\mathbf{l_i}$ [see Niemeier, 2002, p.100].

Stochastic model
The stochastic model describes the probabilistic properties of the observations [see Mikhail and Ackermann, 1982, p.5]. The errors contained in observations are propagated to all quantities computed by the observations. The standard deviation of a single observation can be represented in the variance covariance matrix $\mathbf{\Sigma_{LL}}$ [see Niemeier, 2002, p.106].

$$\mathbf{\Sigma_{LL}} \begin{bmatrix} \sigma_{l_1}^2 & \sigma_{l_1,l_2} & \dots & \sigma_{l_1,l_n} \\ \sigma_{l_2,l_1} & \sigma_{l_2}^2 & \dots & \sigma_{l_2,l_n} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{l_m,l_1} & \sigma_{l_m,l_2} & \dots & \sigma_{l_n}^2 \end{bmatrix} \text{ with } i = 1...n, j = 1...m \tag{39}$$

where $\sigma_{l_i}^2$ are the variances of the observations and $\sigma_{l_i l_j}$ the covariances. If the observations are not correlated, only the diagonal of $\mathbf{\Sigma_{LL}}$ contains the variances $\sigma_{l_i}^2$, the covariances $\sigma_{l_i l_j}$ become zero [see Niemeier, 2002, p.106].

General law of error propagation
The general law of error propagation estimates the magnitude of the propagated error. For a set of m linear functions $f_1, f_2, ...f_m$ with the solution vector x the general law of error propagation is defined as [Ghilani, 2011, p.88]:

$$\mathbf{\Sigma_{xx}} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \dots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix} \begin{bmatrix} \sigma_{l_1}^2 & \sigma_{l_1,l_2} & \dots & \sigma_{l_1,l_n} \\ \sigma_{l_2,l_1} & \sigma_{l_2}^2 & \dots & \sigma_{l_2,l_n} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{l_m,l_1} & \sigma_{l_m,l_2} & \dots & \sigma_{l_n}^2 \end{bmatrix} \times \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \dots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix} \tag{40}$$

in matrix Notation

$$\mathbf{\Sigma_{xx}} = \mathbf{A\Sigma_{LL}A}^T \tag{41}$$

where A is the matrix of coefficients and Σ_{LL} the variance covariance matrix of the observations $l_1, l_2, ..., l_n$.

For non-linear systems the matrix of partial derivatives J (Jacobian matrix) of the m non-linear functions $f_1, f_2, ...f_m$, with respect to the unknown parameters $x_1, x_2, ...x_n$ replaces the matrix of coefficients [Ghilani, 2011, p.88-89].

$$\mathbf{\Sigma_{xx}} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \dots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \dots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \dots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \frac{\partial f_m}{\partial x_2} & \dots & \frac{\partial f_m}{\partial x_n} \end{bmatrix} \begin{bmatrix} \sigma_{l_1}^2 & \sigma_{l_1,l_2} & \dots & \sigma_{l_1,l_n} \\ \sigma_{l_2,l_1} & \sigma_{l_2}^2 & \dots & \sigma_{l_2,l_n} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{l_m,l_1} & \sigma_{l_m,l_2} & \dots & \sigma_{l_n}^2 \end{bmatrix} \times \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \dots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \dots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \dots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \frac{\partial f_m}{\partial x_2} & \dots & \frac{\partial f_m}{\partial x_n} \end{bmatrix} \tag{42}$$

and in matrix notation

$$\mathbf{\Sigma_{xx}} = \mathbf{J\Sigma_{LL}J}^T \tag{43}$$

Adjustment Calculation
Whether an adjustment problem exits or not depends on the ratio of the number of observations n and the number of unknown parameters u. Three cases are distinguished [see Niemeier, 2002, p.100]:

1. 1 $n > u$ overdetermined
2. 2 $n = u$ one explicit solution
3. 3 $n < u$ underdetermined

If the number of observations n and number of unknowns u is equal, exactly one solution exists. In this case, false observations cannot be detected and they will influence the result. In case the system is underdetermined, no total solution can be computed. In case the system is overdetermined, an adjustment problem exists and equation 38 is extended to [Niemeier, 2002]

$$\mathbf{l_i} + \mathbf{v_i} = f_i(X, U) \tag{44}$$

with the residuals $\mathbf{v_i}$. The redundancy r of the system is defined as

$$r = n - u \tag{45}$$

Least squares
To solve the adjustment problem as stated in equation 44, least squares minimizes the squared sum of weighted residuals [Niemeier, 2002].

$$\mathbf{v}^T\mathbf{P}\mathbf{v} \rightarrow min \tag{46}$$

Where $\mathbf{P}$ is the weight matrix [see Niemeier, 2002, p.106]

$$\mathbf{P} = {\mathbf{Q_{LL}}}^{-1} \text{ with } \mathbf{Q_{LL}} = \frac{1}{\sigma_0^2}\mathbf{\Sigma_{LL}} \tag{47}$$

σ_0^2 is the theoretical variance and is often chosen to be equal to 1 [Niemeier, 2002, p.106]. Least squares is described in several textbooks [Niemeier, 2002, Ghilani, 2011, Mikhail and Ackermann, 1982, Neitzel, 2010] and therefore is not described in more detail.

Accuracy measures
All accuracy parameters of the unknowns can be derived from the cofactor matrix $\mathbf{Q_{xx}}$ [Niemeier, 2008]. Considering the Gauss-Markov model the cofactor matrix of the solution vector $\hat{\mathbf{x}}$ can be computed following the general law of error propagation. Let $\hat{\mathbf{x}}$ be the solution vector for the least squares. The cofactor matrix $\mathbf{Q_{xx}}$ for $\hat{\mathbf{x}}$ can be computed with [Niemeier, 2002, p.120]

$$\mathbf{Q_{xx}} = (\mathbf{A}^T\mathbf{P}\mathbf{A})^{-1} \tag{48}$$

The empirical reference standard deviation s_0 is defined as [Niemeier, 2008, p.272]:

$$s_0 = \sqrt{\frac{\mathbf{v}^T\mathbf{P}\mathbf{v}}{n-u}} \tag{49}$$

From $\mathbf{Q_{xx}}$ and s_0 the variance covariance matrix of the adjusted unknowns $\hat{\mathbf{S}}_{\mathbf{xx}}$ can be derived as follows [Niemeier, 2008, p.272]:

$$\hat{\mathbf{S}}_{\mathbf{xx}} = s_0^2 \cdot \mathbf{Q_{xx}} \tag{50}$$

The accuracy of a single unknown parameter is then [Niemeier, 2008, p.274]

$$s_{x_i} = s_0\sqrt{q_{x_i,x_i}} \tag{51}$$

Niemeier [2008] states, that beside the accuracy the parameters of the reliability of an adjustment model yield the possibility to control the observations and their influence on the parameters of the model. All common parameters of the reliability can be derived from the cofactor matrix of the residuals $\mathbf{Q_{vv}}$. The parameters of the reliability reveal for example the magnitude and influence of gross errors [see Niemeier, 2008]. It is distinguished between the parameters of the internal and external reliability. The parameters of the reliability for geodetic networks, their meaning and boundary values applied in practice are explained in detail in the publications of [Niemeier, 2008, Neitzel, 2010], please refer to these for detailed explanations. Both the parameters of the internal and external reliability were calculated for the pose estimation of the BSM. As the adjustment present in this thesis differs from the adjustment problems of geodetic networks, the interpretation of the parameters is difficult, it is beyond the scope of this thesis.

C Error Estimation

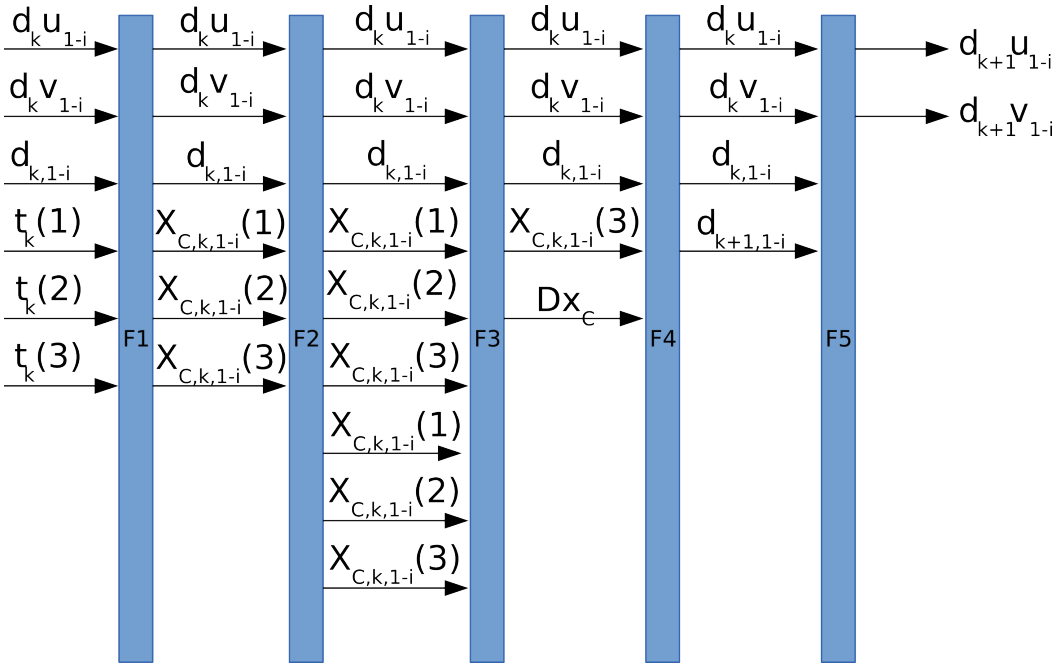


Figure 61: Error propagation flowchart: The error is propagated through the set of functions of five functional models. **F1** to **F5** are the corresponding design matrices. The error is propagated from the left to the right. The parameters on the arrows are the input and output parameters respectively. This propagation is applied iteratively at each iteration of the pose estimation algorithm.

The error is propagated iteratively through the set of functions of five functional models. For each functional model a design matrix **F1** to **F5** is set up. The design matrices **F1**, **F2**, **F3**, **F4**, **F5** contain the partial derivatives of the functions of the output parameter with respect to the input parameters.

- functional model for design matrix **F1**:

$$\begin{aligned} d_k u &= d_k u \\ d_k v &= d_k v \\ d_k = d_k \mathbf{x_{C_k}} &= \mathbf{R}[\mathbf{I}|\mathbf{x_W}]\mathbf{t_k} \end{aligned} \tag{52}$$

- functional model for design matrix **F2**:

$$\begin{aligned} d_k u &= d_k u \\ d_k v &= d_k v \\ d_k &= d_k \\ \mathbf{x_C} &= \mathbf{x_C} \\ \bar{x_{C_k}}(1) &= \frac{1}{N}\sum_{i=1}^{i=N}(x_{C_k}(1)) \\ \bar{x_{C_k}}(2) &= \frac{1}{N}\sum_{i=1}^{i=N}(x_{C_k}(2)) \\ \bar{x_{C_k}}(3) &= \frac{1}{N}\sum_{i=1}^{i=N}(x_{C_k}(3)) \end{aligned} \tag{53}$$

- functional model for design matrix **F3**:

$$\begin{aligned} d_k u &= d_k u \\ d_k v &= d_k v \\ d_k &= d_k \\ x_C(3) &= x_C(3) \\ D_{x_C} &= \sum_{n=1}^{N}(||x_C - \bar{x_C}||^2) \end{aligned} \tag{54}$$

- functional model for design matrix **F4**:

$$\begin{aligned} d_k u &= d_k u \\ d_k v &= d_k v \\ d_k &= d_k \\ d_{k+1} &= \left(\frac{D_{x_W}}{D_{x_C}} x_{C_k}(3)\right) \end{aligned} \tag{55}$$

- functional model for design matrix **F5**:

$$\begin{aligned} d_{k+1} u &= \frac{d_k u}{d_k d_{k+1}} \\ d_{k+1} v &= \frac{d_k v}{d_k dk + 1} \end{aligned} \tag{56}$$

The final variance covariance matrix of the unknowns $d_k u, d_k v$ is than computed with:

$$\mathbf{\Sigma_{xx}} = \mathbf{F} \cdot \mathbf{\Sigma_{LL}} \cdot \mathbf{F}^T \text{ with: } \mathbf{F} = \mathbf{F5} \cdot \mathbf{F4} \cdot \mathbf{F3} \cdot \mathbf{F2} \cdot \mathbf{F1} \tag{57}$$

The design matrices **F1** to **F5** are multiplied and the final variance covariance matrix of the unknowns is calculated following the general law of error propagation. $\mathbf{\Sigma_{LL}}$ is the variance covariance matrix of the observations, that are the 2D image coordinates of the matched shadow centroids of the descent-image. This error propagation did not fully take the convergence of the pose estimation algorithm. An improvement of the approach was beyond the scope of the thesis.

D Preprocessing of the high-resolution DEMs

From a x,y,z point cloud in a local coordinate system a point cloud in the moon-fixed coordinate system is derived. The location of the landing site on the moon is set to latitude $\phi = -5°$, longitude $\lambda = 8°$, which is a possible landing site location for real missions. The x, y, z point coordinates in the local coordinate system are given in millimeters.

In order to geo-reference the DEM three transformation are necessary. First the coordinates are converted to meter and the point cloud is scaled by the factor 100. Both is done with the homogeneous transform

$$\mathbf{T_{scale}} = \begin{bmatrix} 100/10000 & 0 & 0 & 0 \\ 0 & 100/10000 & 0 & 0 \\ 0 & 0 & 100/10000 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{58}$$

Second the origin of the local coordinate system is shifted to point $\mathbf{p_m}$, which is located in the middle of the point cloud:

$$\mathbf{T^{local}_{middle}} = \begin{bmatrix} 1 & 0 & 0 & p_{m_x} \\ 0 & 1 & 0 & p_{m_y} \\ 0 & 0 & 1 & p_{m_z} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{59}$$

Finally, the point cloud is transformed to the moon-fixed coordinate system by shifting and rotating $\mathbf{p_m}$ in a way, that the point cloud is located at $\phi = -5°$, $\lambda = 8°$ and is tangential to the lunar surface. As the point cloud has only an extent of 500 m x 600 m the moon curvature is neglected. The transformation $\mathbf{T^{middle}_{moon}}$ is constructed as follows:

$$\begin{aligned} \mathbf{T^{middle}_{moon}} &= \begin{bmatrix} \mathbf{a} & \mathbf{b} & \mathbf{c} \end{bmatrix} \text{ where} \\ \mathbf{c} &= \begin{bmatrix} \frac{e_1}{|e|} \\ \frac{e_2}{|e|} \\ \frac{e_3}{|e|} \end{bmatrix} \text{ and} \\ \mathbf{a'} &= \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \text{ or } \mathbf{a'} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \text{ if } \mathbf{c} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \text{ then} \\ \mathbf{b} &= \mathbf{a'} \times \mathbf{c} \text{ and} \\ \mathbf{a} &= \mathbf{c} \times \mathbf{b} \end{aligned} \tag{60}$$

where $\mathbf{e}$ is the translation vector to the point $\phi = -5°$, $\lambda = 8°$ on the lunar surface. The concatenation of the three transformations leads to

$$\mathbf{T_{pointCloud}} = \mathbf{T^{middle}_{moon}} \cdot \mathbf{T^{local}_{middle}} \cdot \mathbf{T_{scale}} \tag{61}$$

The transformation of the point cloud with $\mathbf{T_{pointCloud}}$ results in a geo-referenced 3D point cloud in the moon-fixed coordinate system.

E Structure of a Boxplot

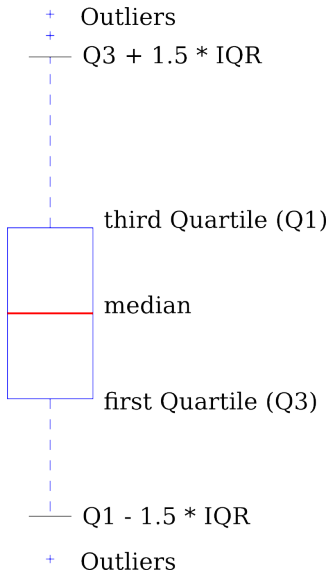


Figure 62: The box of a boxplot draws the data in the range of the first Quartile (Q1) and the third Quartile (Q3). The red line indicates the median of the data. The Inter Quartile Range (IQR) is given as Q1-Q3. The top whisker ends at Q3+1.5*IQR and the bottom whisker at Q1-1.5*IQR. The cross above and below the top and bottom whiskers indicate outliers.

F Experiment results

F.1 Results experiment 1 landing site 1

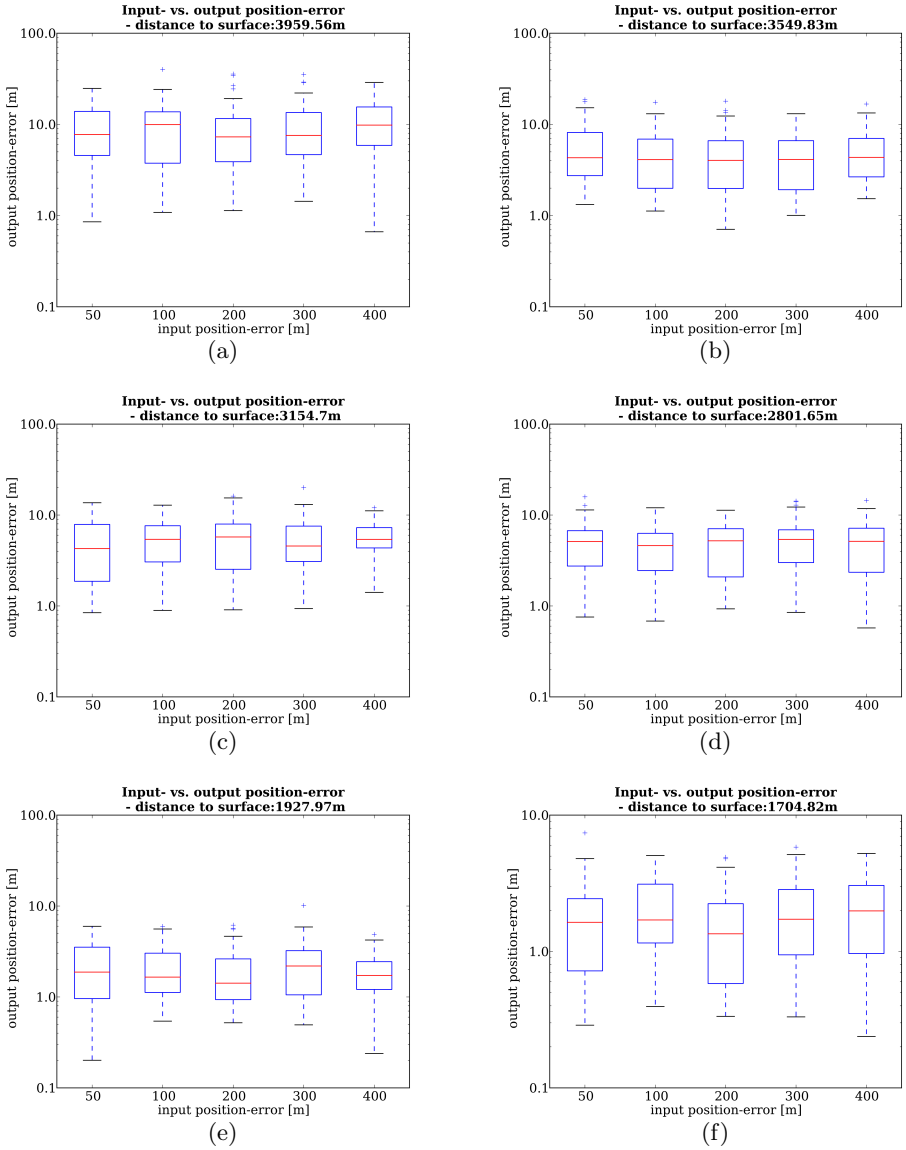


Figure 63: Error vs. initial offset of the trajectory offset - landing site 1 (4km - 1.7km) : Box plots showing the error of the estimated position with respect to the initial offset of 50 m, 100 m, 200 m, 300 m and 400 m for a certain surface distance.

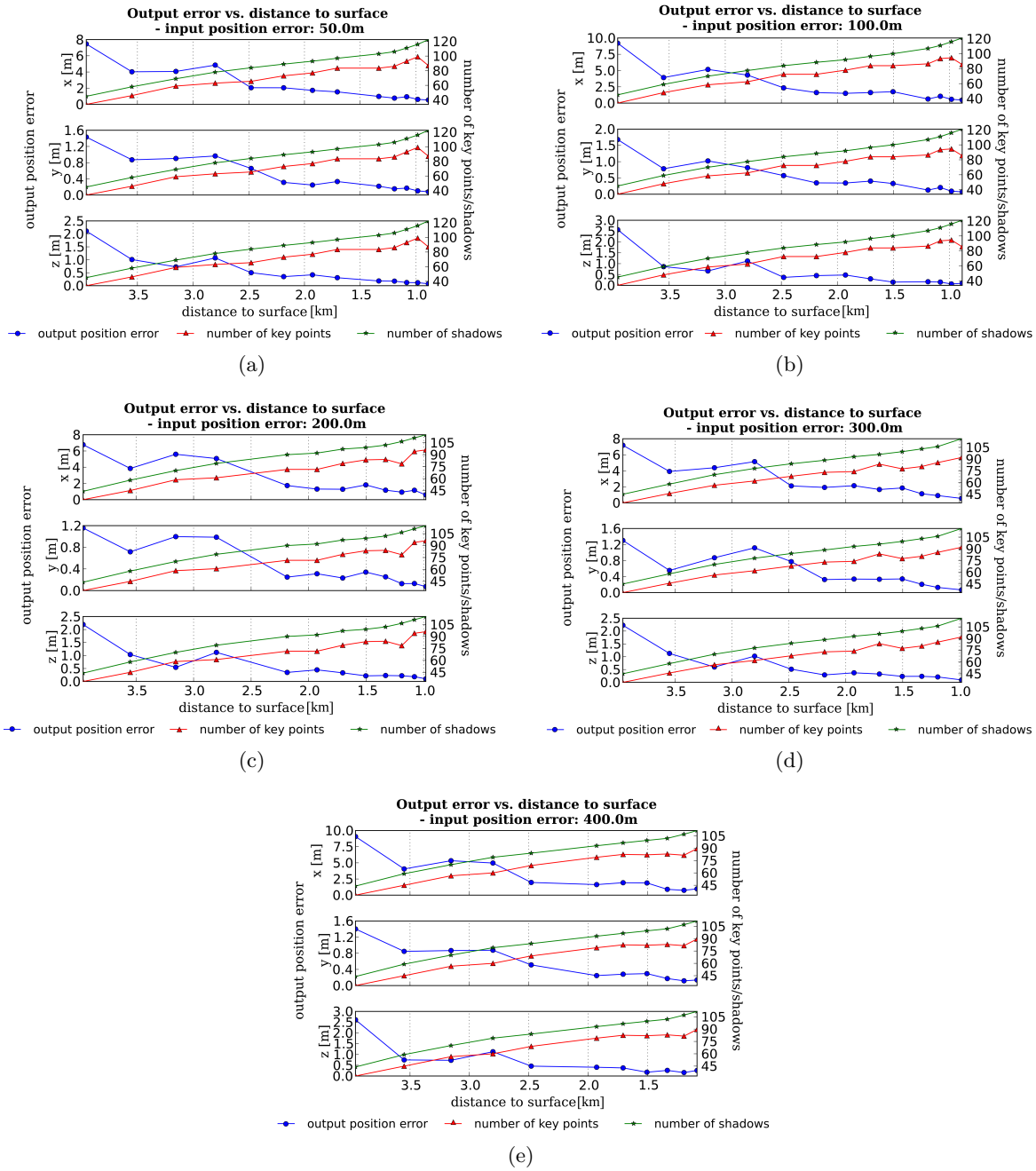


Figure 64: Trajectory error landing site 1 - experiment 1 (initial offset 100, 200, 300, 400m): Plot of the error of the estimated position, the number of shadows and the number of key points with respect to the distance to the surface. The left y-axis relates to the output position error (blue) and the right y-axis relates to the number of shadows (green) and the number of key points used to estimate the position (red).

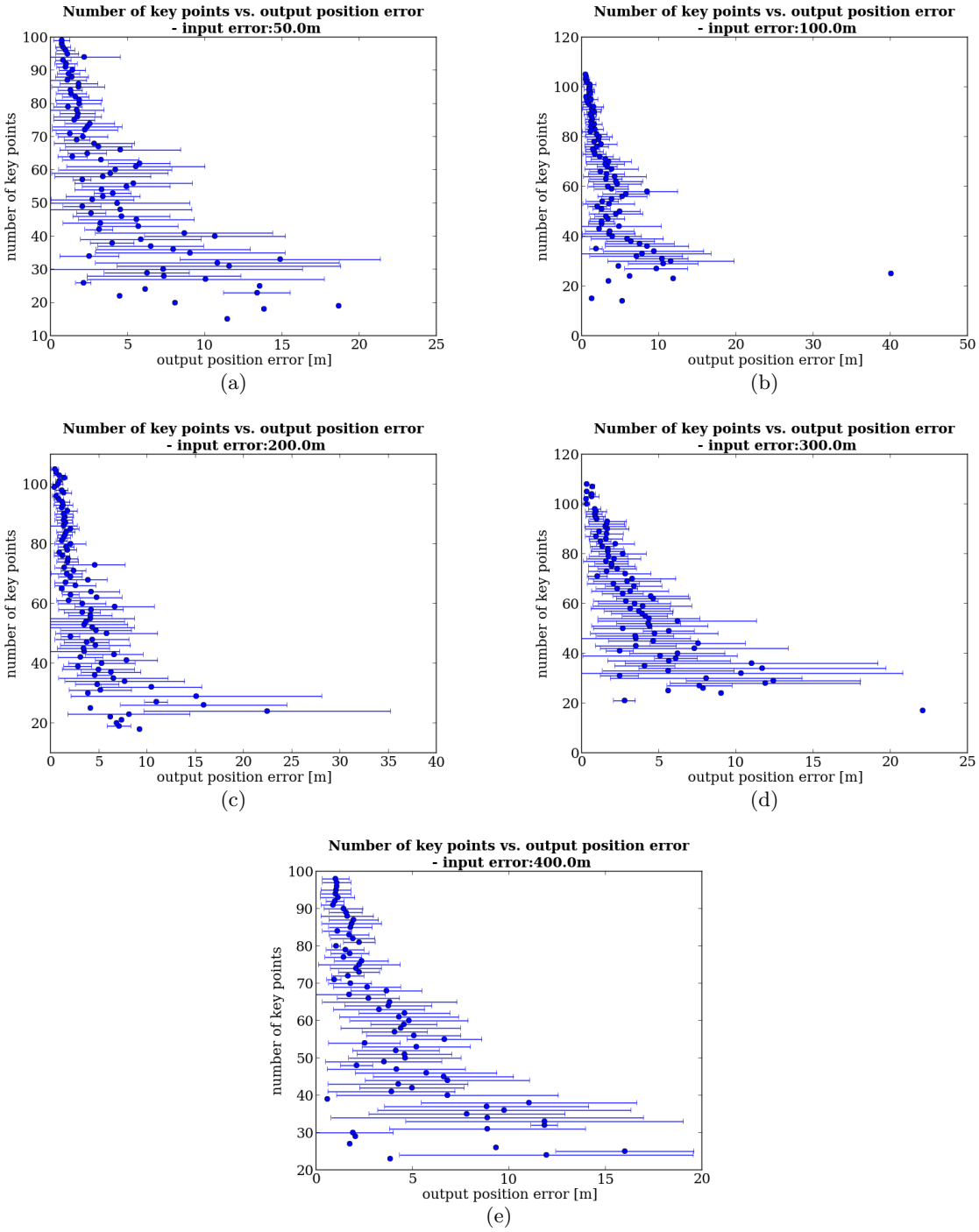
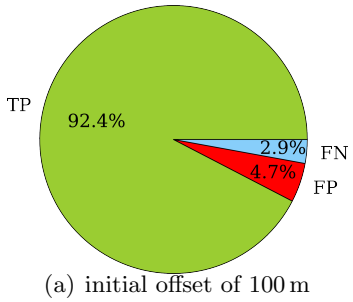


Figure 65: Number of key points vs. error estimated position for landing site 1 (initial offset 100, 200, 300, 400 m): Plot of the number of key points with respect to the error of the estimated position. The blue points mark the mean value of the output error and the whiskers show the standard deviation $\pm\sigma$. Points with no whiskers imply, that only one pose estimation was done with this amount of key points.

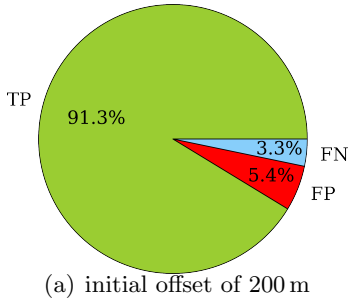


(a) initial offset of 100 m

	True matches	True non-matches		
Predicted matches	TP= 53055	FP= 2719	P` = 55774	PVV= 0.95
Predicted non-matches	FN= 1647	TN= 0	N` = 1647	
	P = 54702	N = 2719	Total = 57421	
	TPR = 0.97	FPR = 1.0		ACC = 0.92

(b)

Figure 66: Confusion matrix - trajectory offset of 100 m - landing site 1 - experiment 1 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations of experiment 1 with an initial trajectory offset of 100 m.

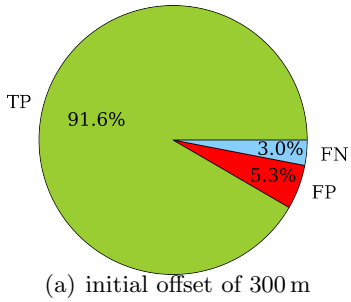


(a) initial offset of 200 m

	True matches	True non-matches		
Predicted matches	TP= 48415	FP= 2887	P` = 36401	PVV= 0.94
Predicted non-matches	FN= 1726	TN= 0	N` = 9811	
	P = 50141	N = 378	Total = 46212	
	TPR = 0.97	FPR = 1.0		ACC = 0.91

(b)

Figure 67: Confusion matrix - trajectory offset of 200 m - landing site 1 - experiment 1 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations of experiment 1 with an initial trajectory offset of 200 m.

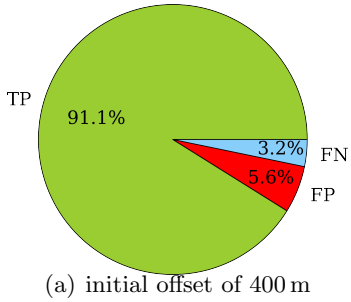


(a) initial offset of 300 m

	True matches	True non-matches		
Predicted matches	TP= 47291	FP= 2753	P` = 50044	PVV= 0.94
Predicted non-matches	FN= 1566	TN= 0	N` = 1566	
	P = 48857	N =2753	Total = 51610	
	TPR = 0.97	FPR = 1.0		ACC = 0.91

(b)

Figure 68: Confusion matrix - trajectory offset of 300 m - landing site 1 - experiment 1 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations of experiment 1 with an initial trajectory offset of 300 m.



(a) initial offset of 400 m

	True matches	True non-matches		
Predicted matches	TP= 42744	FP= 2642	P` = 45386	PVV= 0.94
Predicted non-matches	FN= 1514	TN= 0	N` = 1514	
	P = 44258	N =2642	Total = 46900	
	TPR = 0.97	FPR = 1.0		ACC = 0.91

(b)

Figure 69: Confusion matrix - trajectory offset of 400 m - landing site 1 - experiment 1 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations of experiment 1 with an initial trajectory offset of 400 m.

F.2 Results experiment 1 landing site 2

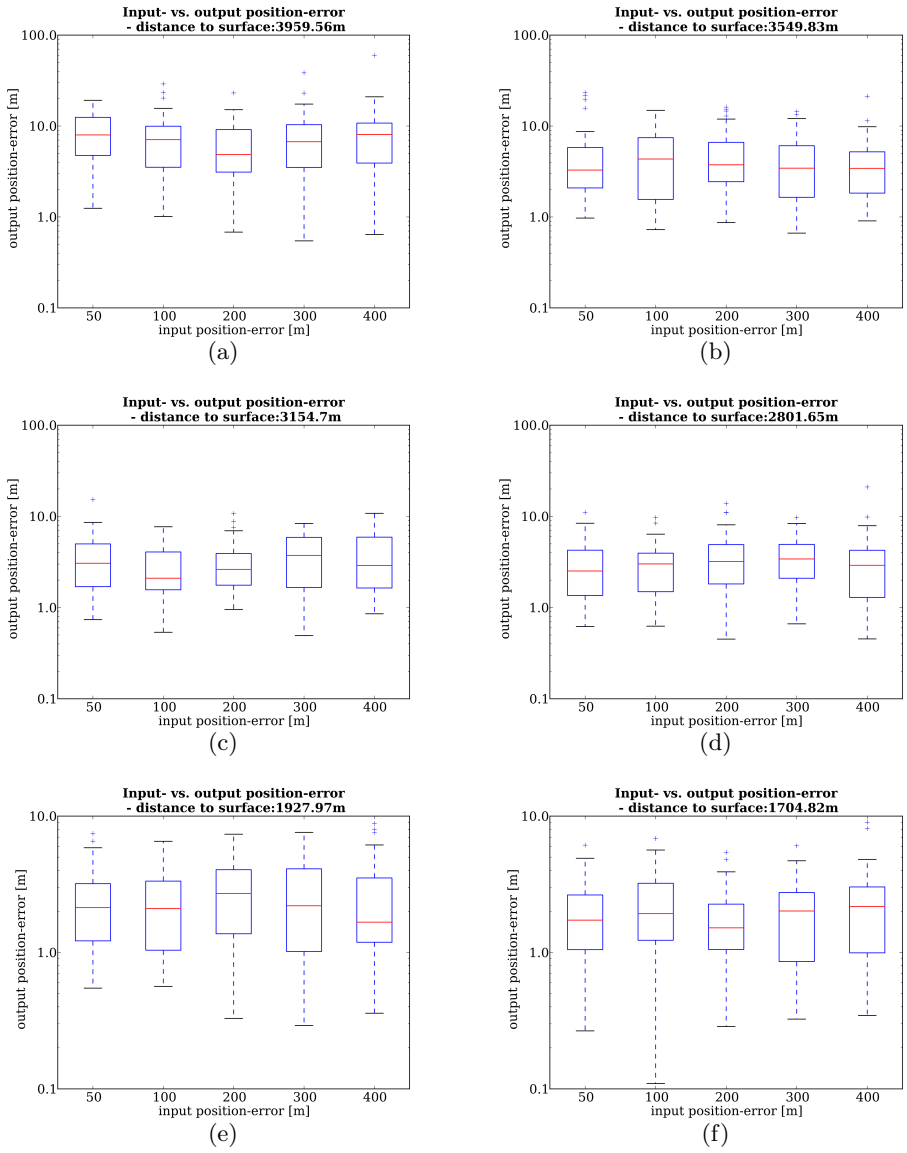


Figure 70: Error vs. initial offset of the trajectory offset - landing site 2 (4km - 1.7km) : Box plots showing the error of the estimated position with respect to the initial offset of 50 m, 100 m, 200 m, 300 m and 400 m for a certain surface distance.

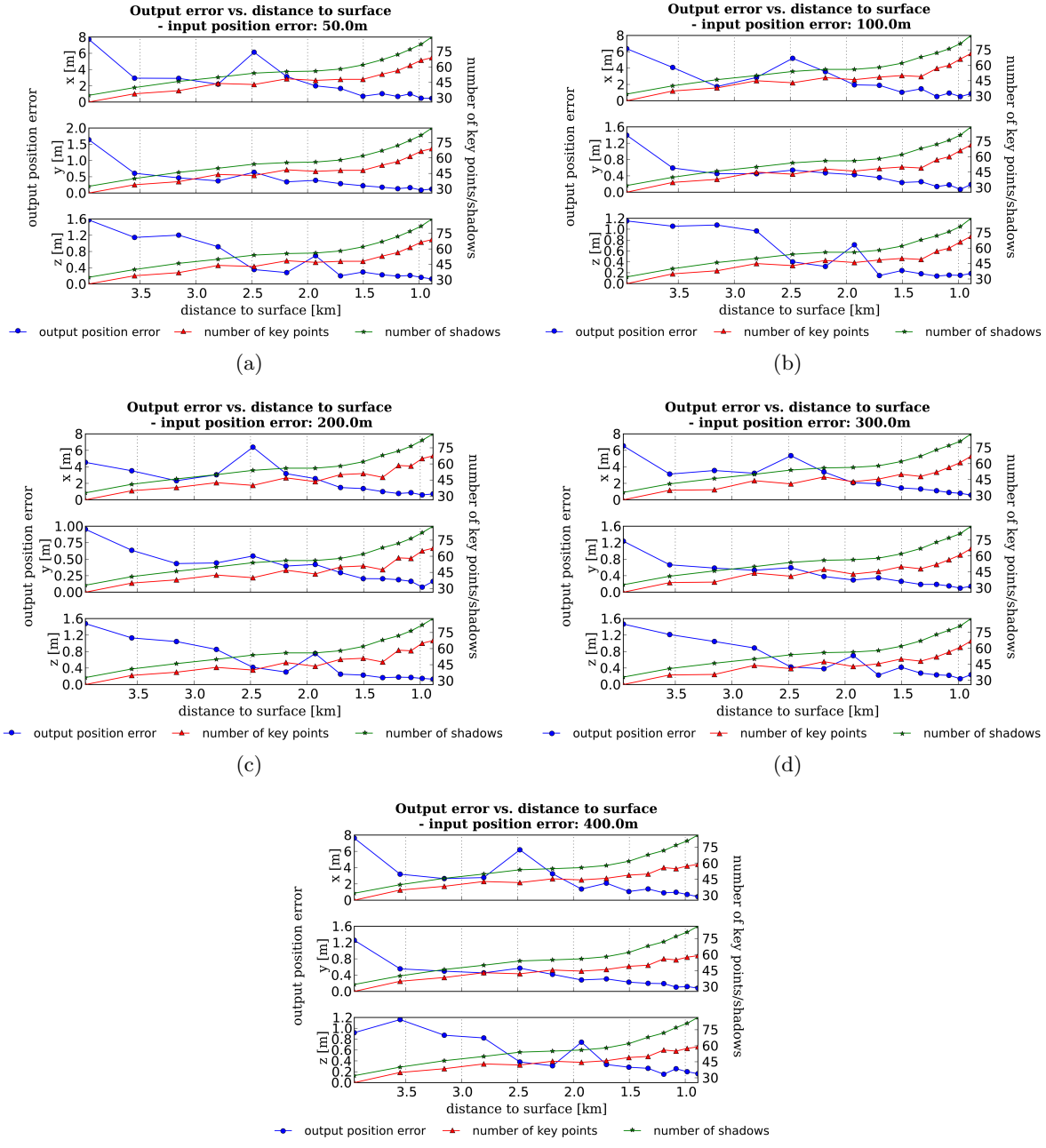


Figure 71: Trajectory error landing site 2 - experiment 1 (initial offset 100, 200, 300, 400m): Plot of the error of the estimated position, the number of shadows and the number of key points with respect to the distance to the surface. The left y-axis relates to the output position error (blue) and the right y-axis relates to the number of shadows (green) and the number of key points used to estimate the position (red).

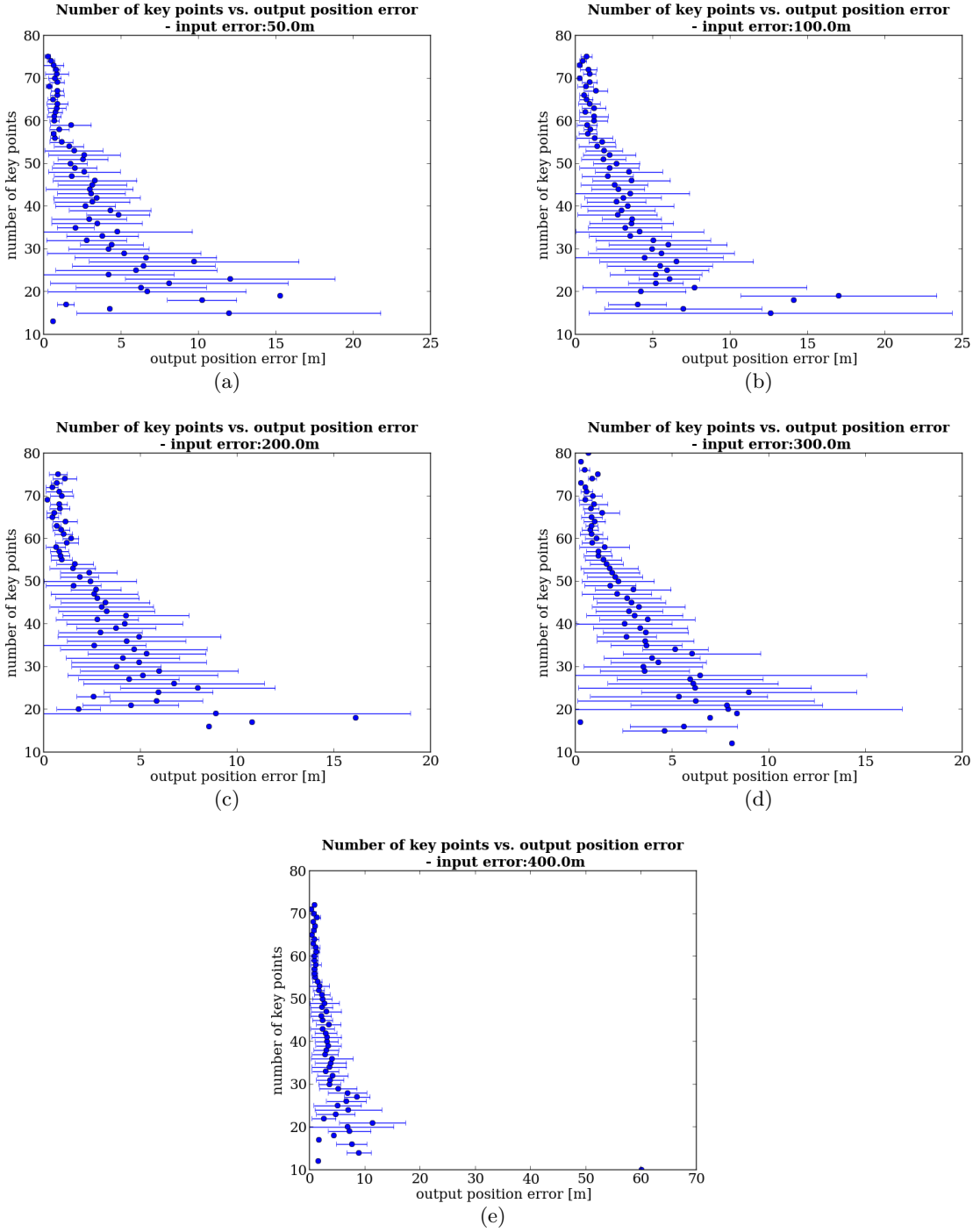
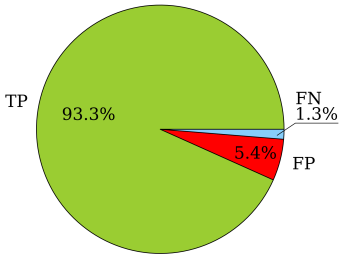


Figure 72: Number of key points vs. error estimated position for landing site 2 (initial offset 100, 200, 300, 400m): Plot of the number of key points with respect to the error of the estimated position. The blue points mark the mean value of the output error and the whiskers show the standard deviation $\pm\sigma$. Points with no whiskers imply, that only one pose estimation was done with this amount of key points.

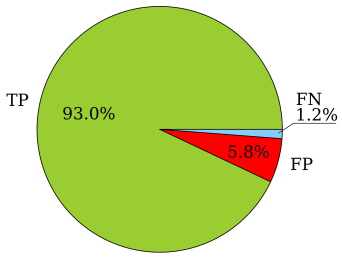


(a) initial offset of 100 m

	True matches	True non-matches		
Predicted matches	TP= 39116	FP= 2264	P` = 41380	PVV= 0.95
Predicted non-matches	FN= 537	TN= 0	N` = 537	
	P = 39653	N =2264	Total = 41917	
	TPR = 0.99	FPR = 1.0		ACC = 0.93

(b)

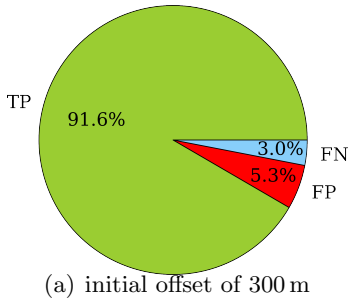
Figure 73: Confusion matrix - trajectory offset of 100 m - landing site 2 - experiment 1 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations of experiment 1 with an initial trajectory offset of 100 m.



(a) initial offset of 200 m

	True matches	True non-matches		
Predicted matches	TP= 38986	FP= 2433	P` = 41419	PVV= 0.99
Predicted non-matches	FN= 508	TN= 0	N` = 508	
	P = 39494	N =2433	Total = 41927	
	TPR = 0.75	FPR = 1.0		ACC = 0.74

Figure 74: Confusion matrix - trajectory offset of 200 m - landing site 2 - experiment 1 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations of experiment 1 with an initial trajectory offset of 200 m.

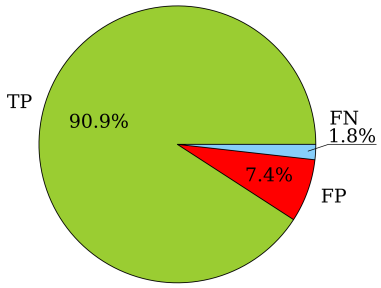


(a) initial offset of 300 m

	True matches	True non-matches		
Predicted matches	TP= 38804	FP= 2527	P` = 41331	PVV= 0.94
Predicted non-matches	FN= 573	TN= 0	N` = 573	
	P = 39377	N =2527	Total = 41904	
	TPR = 0.99	FPR = 1.0		ACC = 0.93

(b)

Figure 75: Confusion matrix - trajectory offset of 300 m - landing site 2 - experiment 1 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations of experiment 1 with an initial trajectory offset of 300 m.



(a) initial offset of 400 m

	True matches	True non-matches		
Predicted matches	TP= 38060	FP= 3087	P` = 41138	PVV= 0.92
Predicted non-matches	FN= 744	TN= 0	N` = 744	
	P = 38804	N =3087	Total = 41882	
	TPR = 0.98	FPR = 1.0		ACC = 0.91

Figure 76: Confusion matrix - trajectory offset of 400 m - landing site 2 - experiment 1 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations of experiment 1 with an initial trajectory offset of 400 m.

F.3 Results experiment 1 landing site 3

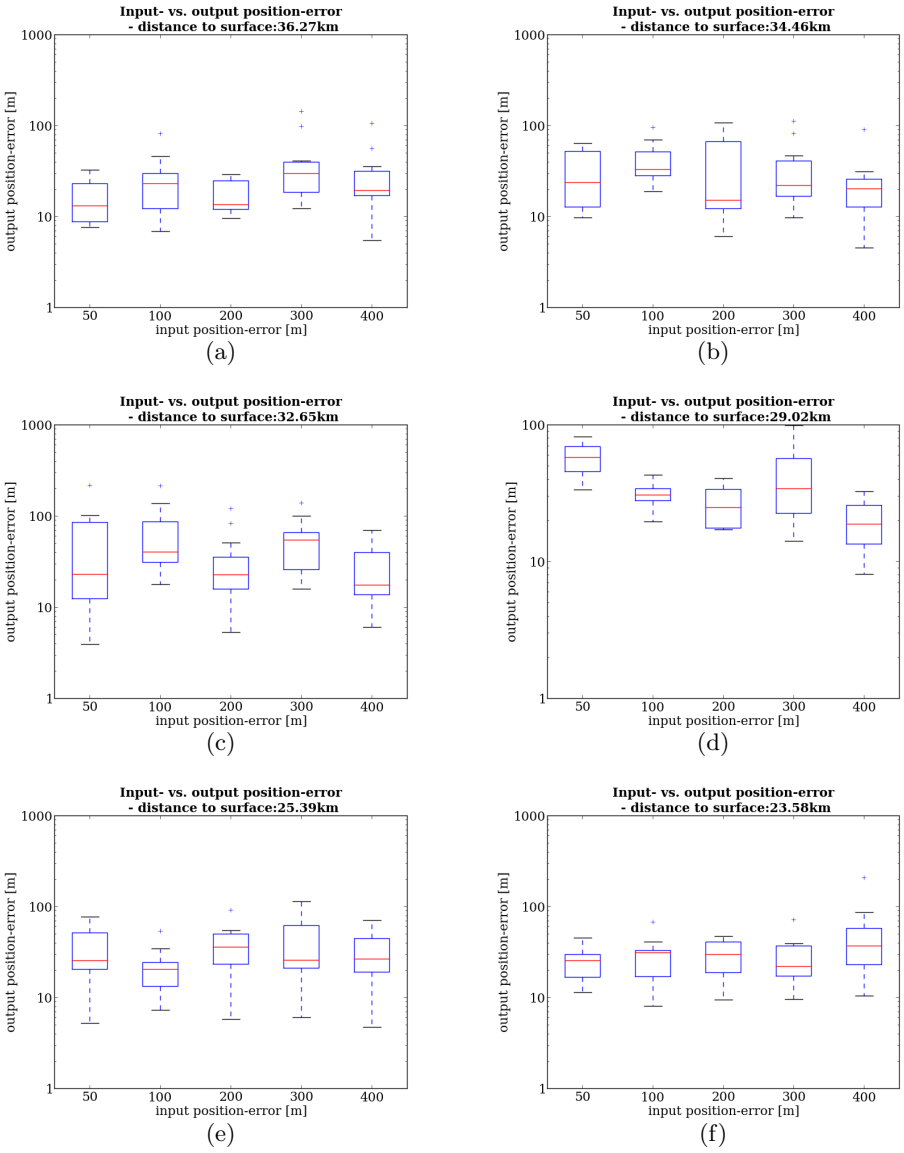


Figure 77: Error vs. initial offset of the trajectory offset - landing site 3 (36km - 23km) : Box plots showing the error of the estimated position with respect to the initial offset of 50 m, 100 m, 200 m, 300 m and 400 m for a certain surface distance.

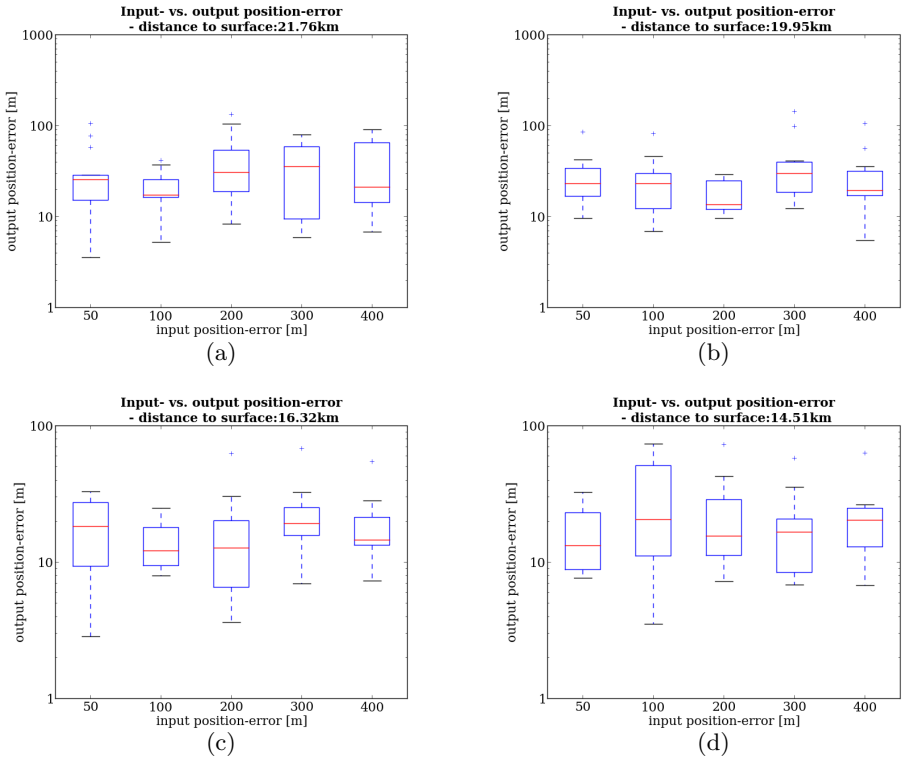


Figure 78: Error vs. initial offset of the trajectory offset - landing site 3 (22km - 15km) : Box plots showing the error of the estimated position with respect to the initial offset of 50 m, 100 m, 200 m, 300 m and 400 m for a certain surface distance.

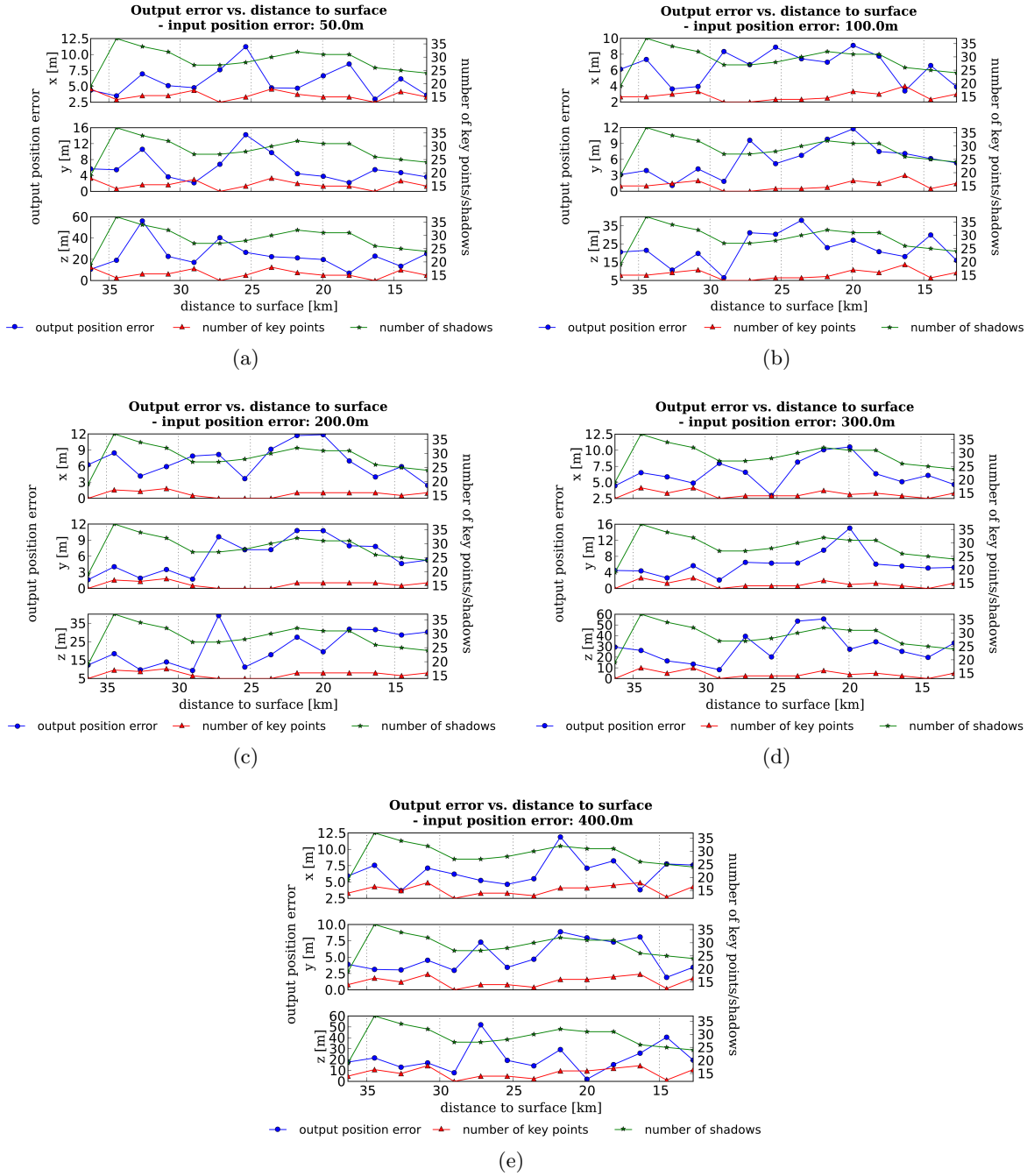


Figure 79: Trajectory error landing site 3 - experiment 1 (initial offset 100, 200, 300, 400m): Plot of the error of the estimated position, the number of shadows and the number of key points with respect to the distance to the surface. The left y-axis relates to the output position error (blue) and the right y-axis relates to the number of shadows (green) and the number of key points used to estimate the position (red).

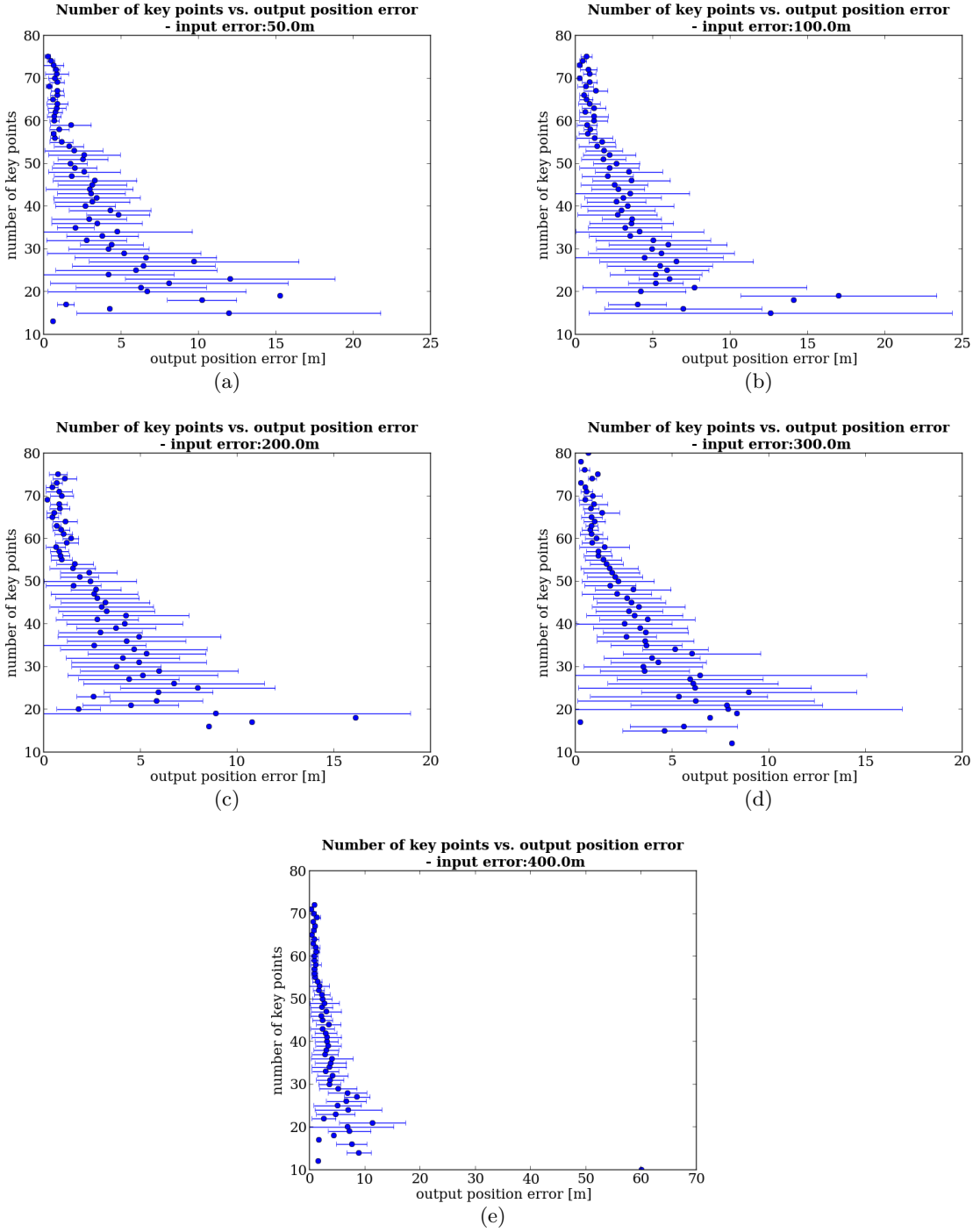
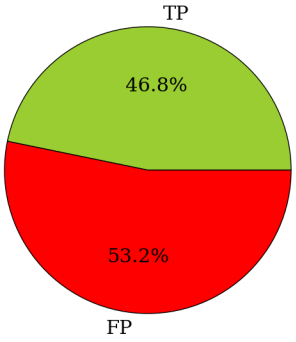


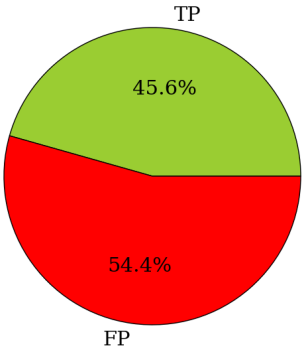
Figure 80: Number of key points vs. error estimated position for landing site 3 (initial offset 100, 200, 300, 400 m): Plot of the number of key points with respect to the error of the estimated position. The blue points mark the mean value of the output error and the whiskers show the standard deviation $\pm\sigma$. Points with no whiskers imply, that only one pose estimation was done with this amount of key points.



(a) initial offset of 100 m

	True matches	True non-matches		
Predicted matches	TP= 2488	FP= 2831	P' = 5319	PVV= 0.47
Predicted non-matches	FN= 0	TN= 0	N' = 0	
	P = 2488	N = 2831	Total = 5319	
	TPR = 1.0	FPR = 1.0		ACC = 0.47

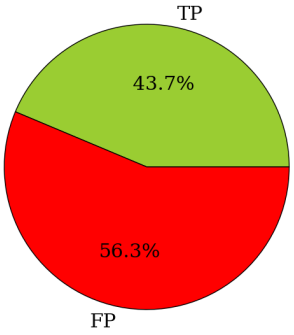
Figure 81: Confusion matrix - trajectory offset of 100 m - landing site 3 - experiment 1 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations of experiment 1 with an initial trajectory offset of 100 m.



(a) initial offset of 200 m

	True matches	True non-matches		
Predicted matches	TP= 2358	FP= 2811	P' = 5169	PVV= 0.46
Predicted non-matches	FN= 0	TN= 0	N' = 0	
	P = 2358	N = 2811	Total = 5169	
	TPR = 1.0	FPR = 1.0		ACC = 0.56

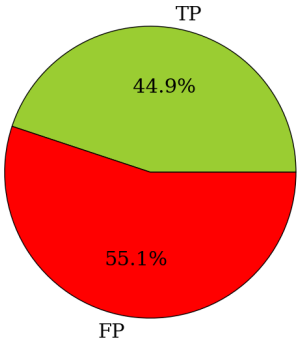
Figure 82: Confusion matrix - trajectory offset of 200 m - landing site 3 - experiment 1 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations of experiment 1 with an initial trajectory offset of 200 m.



(a) initial offset of 300 m

	True matches	True non-matches		
Predicted matches	TP= 2060	FP= 2656	P' = 4716	PVV= 0.44
Predicted non-matches	FN= 0	TN= 0	N' = 0	
	P = 2060	N = 2656	Total = 4716	
	TPR = 1.0	FPR = 1.0		ACC = 0.44

Figure 83: Confusion matrix - trajectory offset of 300 m - landing site 3 - experiment 1 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations of experiment 1 with an initial trajectory offset of 300 m.



(a) initial offset of 400 m

	True matches	True non-matches		
Predicted matches	TP= 1809	FP= 2219	P' = 4028	PVV= 0.45
Predicted non-matches	FN= 0	TN= 0	N' = 0	
	P = 1809	N = 2219	Total = 4028	
	TPR = 1.0	FPR = 1.0		ACC = 0.45

Figure 84: Confusion matrix - trajectory offset of 400 m - landing site 3 - experiment 1 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations of experiment 1 with an initial trajectory offset of 400 m.

F.4 Results experiment 2 landing site 1

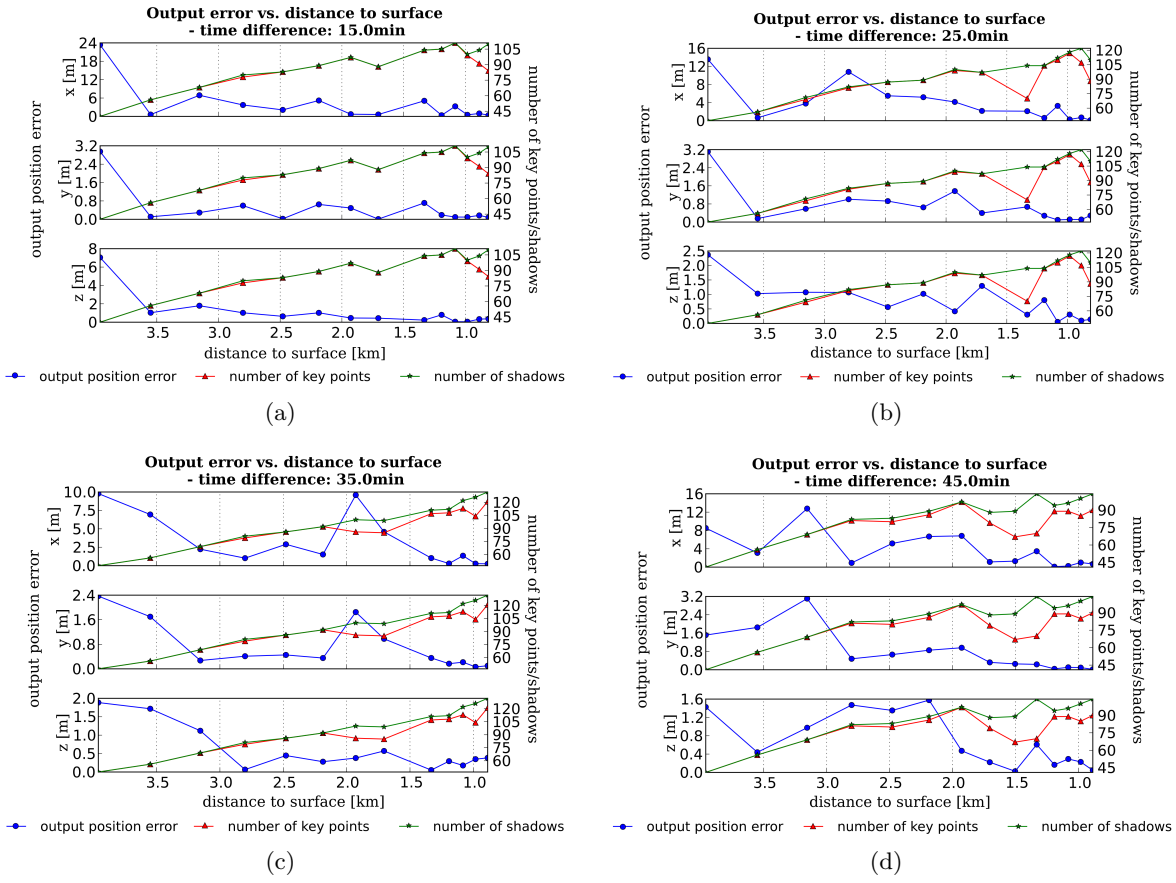
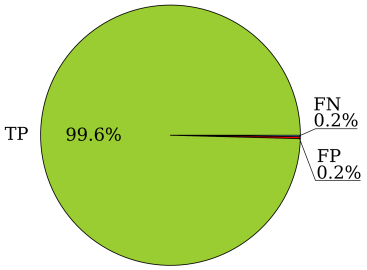


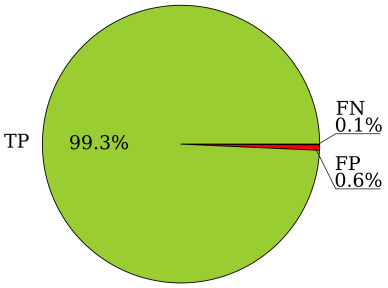
Figure 85: Trajectory error landing site 1 - experiment 2 (15, 25, 35, 45 minutes): Plot of the error of the estimated position, the number of shadows and the number of key points with respect to the distance to the surface. The left y-axis relates to the output position error (blue) and the right y-axis relates to the number of shadows (green) and the number of key points used to estimate the position (red).



(a) difference in time 15 minutes

	True matches	True non-matches		
Predicted matches	TP= 1197	FP= 3	P' = 1200	PVV= 0.99
Predicted non-matches	FN= 2	TN= 0	N' = 2	
	P = 1199	N =3	Total = 1202	
	TPR = 0.99	FPR = 1		ACC = 0.99

Figure 86: Confusion matrix - difference in time 15 minutes - landing site 1 - experiment 2 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations of experiment 2.

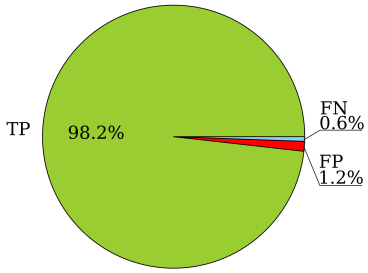


(a) difference in time 25 minutes

.

	True matches	True non-matches		
Predicted matches	TP= 1259	FP= 8	P' = 1267	PVV= 0.99
Predicted non-matches	FN= 1	TN= 0	N' = 1	
	P = 1260	N =8	Total = 1268	
	TPR = 0.99	FPR = 1		ACC = 0.99

Figure 87: Confusion matrix - difference in time 25 minutes - landing site 1 - experiment 2 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations of experiment 2.

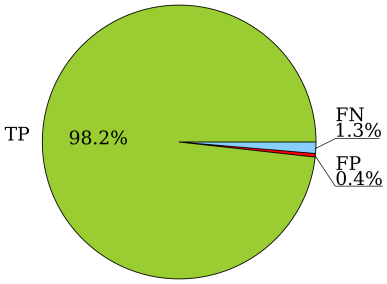


(a) difference in time 35 minutes

	True matches	True non-matches		
Predicted matches	TP= 1217	FP= 15	P' = 1232	PVV= 0.99
Predicted non-matches	FN= 7	TN= 0	N' = 7	
	P = 1224	N = 15	Total = 1239	
	TPR = 0.99	FPR = 1		ACC = 0.98

(b)

Figure 88: Confusion matrix - difference in time 35 minutes - landing site 1 - experiment 2 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations of experiment 2.



(a) difference in time 45 minutes

	True matches	True non-matches		
Predicted matches	TP= 1173	FP= 5	P' = 1178	PVV= 0.99
Predicted non-matches	FN= 16	TN= 0	N' = 16	
	P = 1189	N = 5	Total = 1194	
	TPR = 0.99	FPR = 1		ACC = 0.98

(b)

Figure 89: Confusion matrix - difference in time 45 minutes - landing site 1 - experiment 2 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations of experiment 2.

F.5 Results experiment 2 landing site 2

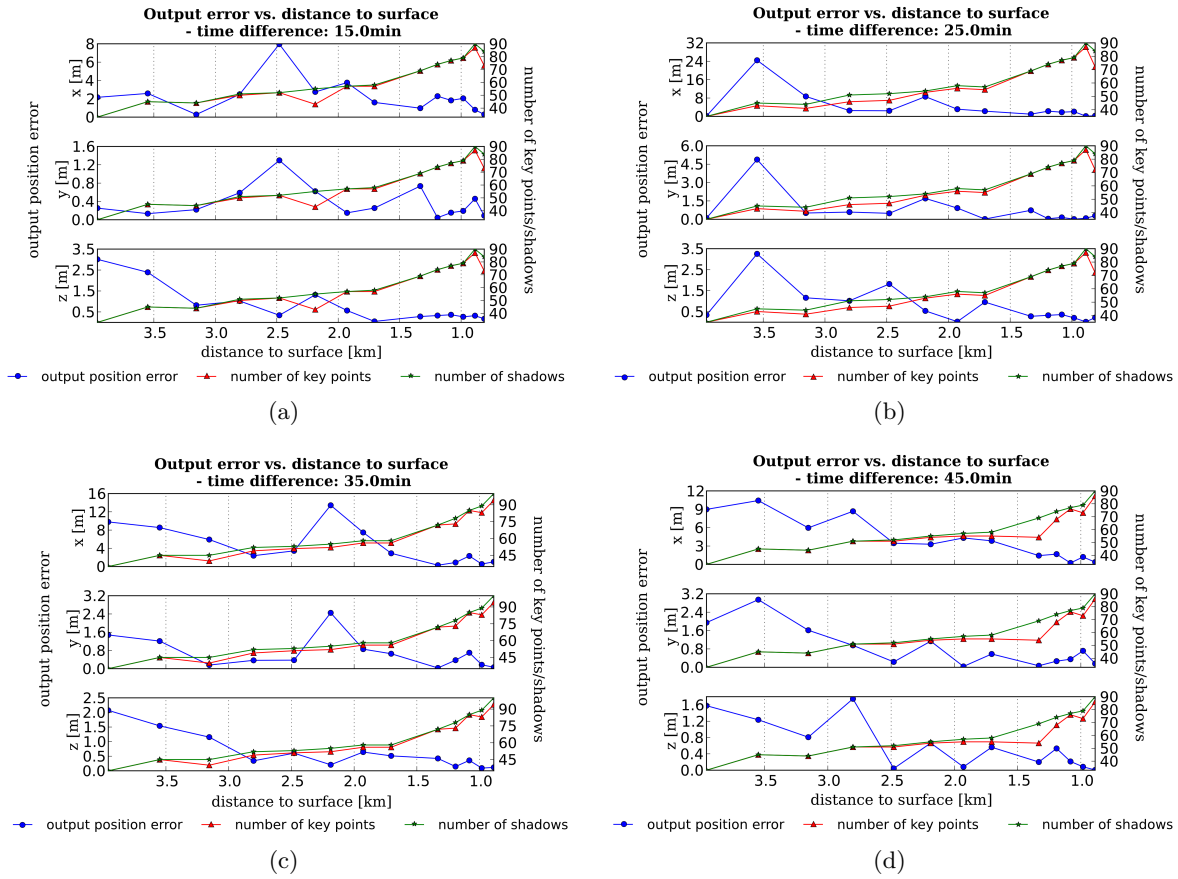
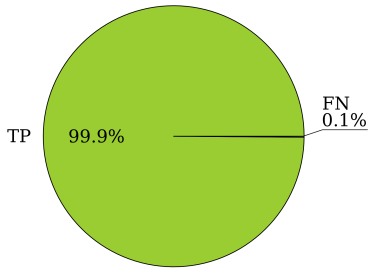


Figure 90: Trajectory error landing site 2 - experiment 2 (15, 25, 35, 45 minutes): Plot of the error of the estimated position, the number of shadows and the number of key points with respect to the distance to the surface. The left y-axis relates to the output position error (blue) and the right y-axis relates to the number of shadows (green) and the number of key points used to estimate the position (red).

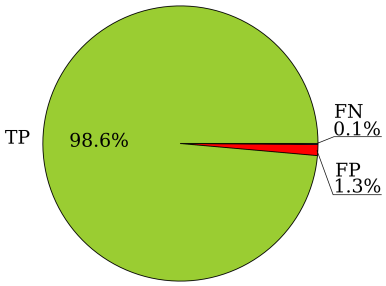


(a) difference in time 15 minutes

	True matches	True non-matches		
Predicted matches	TP= 854	FP= 0	P` = 854	PVV= 1.0
Predicted non-matches	FN= 1	TN= 0	N` = 1	
	P = 855	N =0	Total = 855	
	TPR = 0.99	FPR = 0		ACC = 0.99

(b)

Figure 91: Confusion matrix - difference in time 15 minutes - landing site 2 - experiment 2 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations of experiment 2.

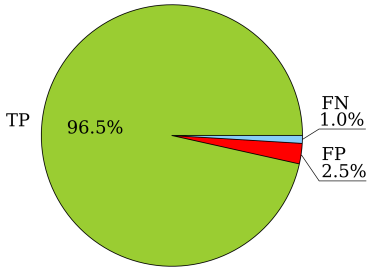


(a) difference in time 25 minutes

	True matches	True non-matches		
Predicted matches	TP= 844	FP= 11	P` = 855	PVV= 0.99
Predicted non-matches	FN= 1	TN= 0	N` = 1	
	P = 845	N = 11	Total = 856	
	TPR = 0.99	FPR = 1		ACC = 0.99

(b)

Figure 92: Confusion matrix - difference in time 25 minutes - landing site 2 - experiment 2 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations of experiment 2.

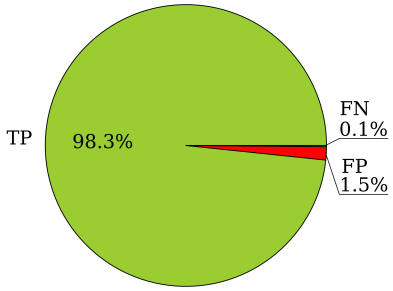


(a) difference in time 35 minutes

	True matches	True non-matches		
Predicted matches	TP= 803	FP= 21	P` = 824	PVV= 0.97
Predicted non-matches	FN= 8	TN= 0	N` = 8	
	P = 811	N = 21	Total = 8832	
	TPR = 0.99	FPR = 1		ACC = 0.97

(b)

Figure 93: Confusion matrix - difference in time 35 minutes - landing site 2 - experiment 2 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations of experiment 2.



(a) difference in time 45 minutes

	True matches	True non-matches		
Predicted matches	TP= 774	FP= 12	P` = 786	PVV= 0.98
Predicted non-matches	FN= 1	TN= 0	N` = 1	
	P = 775	N = 12	Total = 787	
	TPR = 0.99	FPR = 1		ACC = 0.98

(b)

Figure 94: Confusion matrix - difference in time 45 minutes - landing site 2 - experiment 2 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations of experiment 2.

F.6 Results experiment 2 landing site 3

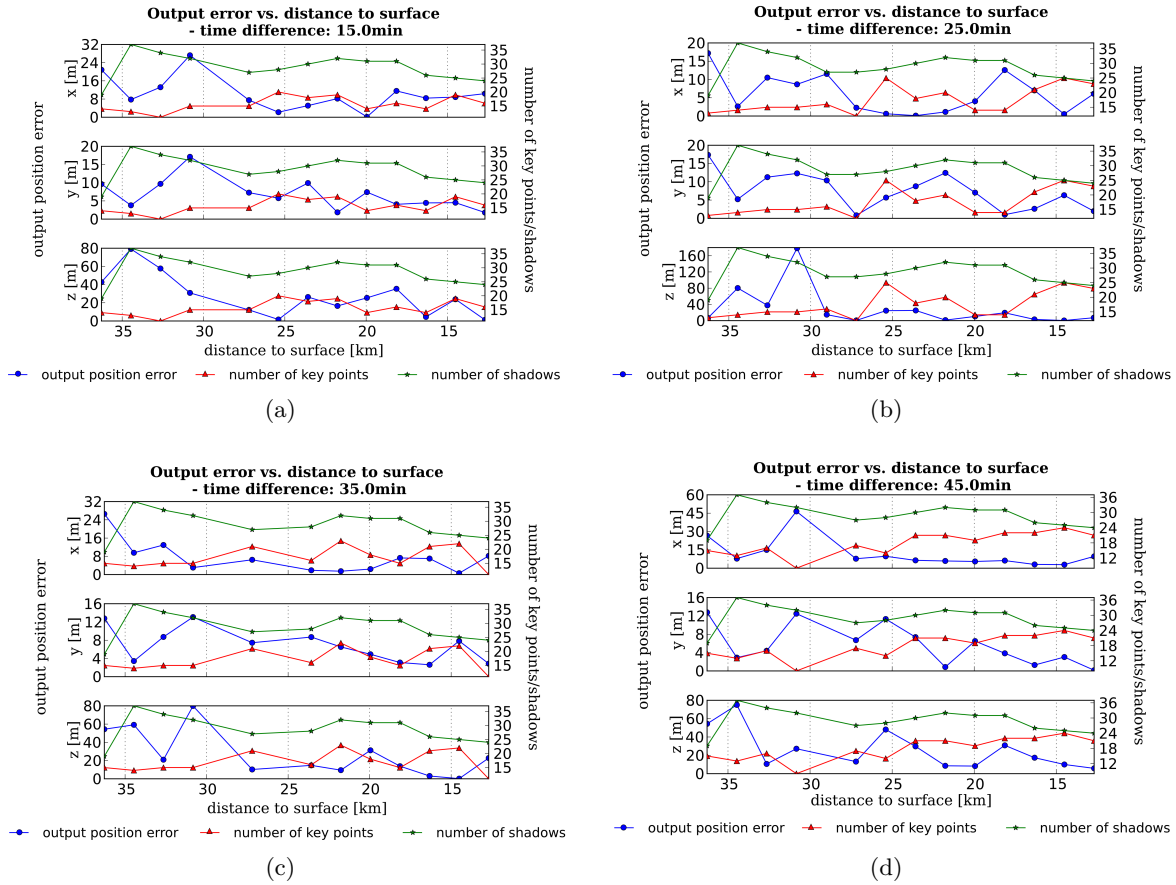
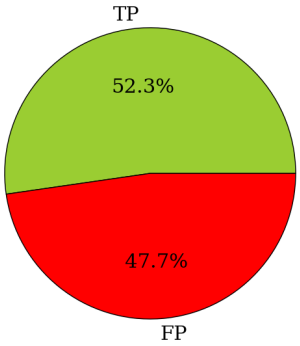


Figure 95: Trajectory error landing site 3 - experiment 2 (15, 25, 35, 45 minutes): Plot of the error of the estimated position, the number of shadows and the number of key points with respect to the distance to the surface. The left y-axis relates to the output position error (blue) and the right y-axis relates to the number of shadows (green) and the number of key points used to estimate the position (red).

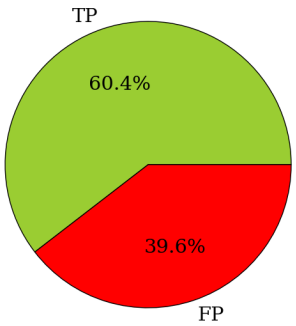


(a) difference in time 15 minutes

	True matches	True non-matches		
Predicted matches	TP= 218	FP = 199	P`= 417	PVV= 0.52
Predicted non-matches	FN= 0	TN= 0	N` = 0	
	P = 218	N = 199	Total = 417	
	TPR = 1.0	FPR = 1.0		ACC = 0.52

(b)

Figure 96: Confusion matrix - difference in time 15 minutes - landing site 3 - experiment 2 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations of experiment 2.

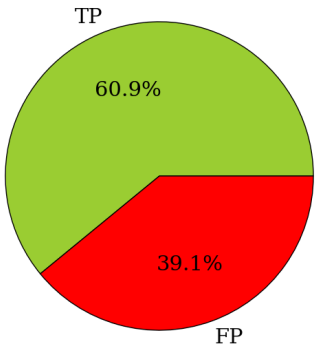


(a) difference in time 25 minutes

	True matches	True non-matches		
Predicted matches	TP= 269	FP = 176	P`= 445	PVV= 0.6
Predicted non-matches	FN= 0	TN= 0	N` = 0	
	P = 269	N = 176	Total = 445	
	TPR = 1.0	FPR = 1.0		ACC = 0.6

(b)

Figure 97: Confusion matrix - difference in time 25 minutes - landing site 3 - experiment 2 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations of experiment 2.

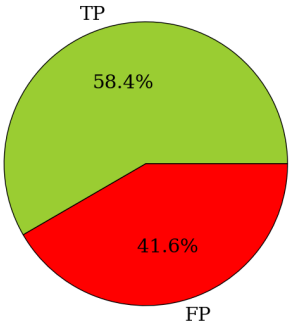


(a) difference in time 35 minutes

	True matches	True non-matches		
Predicted matches	TP= 223	FP = 143	P`= 366	PVV= 0.61
Predicted non-matches	FN= 0	TN= 0	N` = 0	
	P = 223	N = 1143	Total = 366	
	TPR = 1.0	FPR = 1.0		ACC = 0.61

(b)

Figure 98: Confusion matrix - difference in time 35 minutes - landing site 3 - experiment 2 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations of experiment 2.



(a) difference in time 45 minutes

	True matches	True non-matches		
Predicted matches	TP= 234	FP = 167	P`= 401	PVV= 0.58
Predicted non-matches	FN= 0	TN= 0	N` = 0	
	P = 234	N = 167	Total = 401	
	TPR = 1.0	FPR = 1.0		ACC = 0.58

(b)

Figure 99: Confusion matrix - difference in time 45 minutes - landing site 3 - experiment 2 - matching quantity measures: a) visualizes the statistics of TP, FP, TN and FN in a pie chart. b) shows the confusion matrix for all iterations of experiment 2.

References

A. Ansar and K. Daniilidis. Linear Pose Estimation from Points or Lines. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25:282–296, 2003.

A. Appel. Some techniques for shading machine renderings of solids. In *Proceedings of the April 30–May 2, 1968, spring joint computer conference*, AFIPS '68 (Spring), pages 37–45, New York, NY, USA, 1968. ACM.

B. A. Archinal, M. F. AHearn, E. Bowell, A. Conrad, G. J. Consolmagno, R. Courtin, T. Fukushima, D. Hestroffer, J. L. Hilton, G. A. Krasinsky, G. Neumann, J. Oberst, P. K. Seidelmann, P. Stooke, D. J. Tholen, P. C. Thomas, and I. P. Williams. *Report of the IAU Working Group on Cartographic Coordinates and Rotational Elements: 2009.* Springer Science+Business Media B.V., 2010.

D. M. H. Backer, J. W. Head, G. A. Neumann, D. E. Smith, and M. T. Zuber. 7The transition from complex crater to peak-ring basin on the Moon: New observations from the Lunar Orbiter Laser Altimeter (LOLA) instrument. *Journal of Geophysical Research: Planets*, 117(E12):1200, March 2012.

H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-Up Robust Features (SURF). *Comput. Vis. Image Underst.*, 110(3):346–359, June 2008. ISSN 1077-3142.

S. Belongie and J. Malik. Matching with Shape Contexts. *IEEE Workshop on Content-Based Access of Image and Video Libraries*, 2000.

S. Belongie, J. Malik, and J. Puzicha. Shape Context: A new descriptor for shape matching and object recognition. In *NIPS*, pages 831–837, 2000.

S. Belongie, J. Malik, and J. Puzicha. Matching Shapes. *Eight IEEE International Conference on Computer Vision*, 2001.

J.-Y. Bouguet. Pyramidal Implementation of the Lucas Kanade Feature Tracker Description of the algorithm. *Intel Corporation Microprocessor Research Labs*, 2000.

P. A. Brivio, A. Della Ventura, A. Rampini, and R. Schettini. Automatic selection of control-points from shadow structures. *International Journal of Remote Sensing*, 13 (10):1853–1860, 1992.

Y. Cheng and A. Ansar. Landmark Based Position Estimation for Pinpoint Landing on Mars. In *ICRA*, pages 4470–4475. IEEE, 2005.

Y. Cheng, A. E. Johnson, L. H. Matthies, and A. A. Wolf. Passive imaging based hazard avoidance for spacecraft safe landing. *presented at the Int. Symp. Artif. Intell., Robot., Autom. Space*, 2001.

F. C. Crow. SHADOW ALGORITHMS FOR COMPUTER GRAPHICS. *SIGGRAPH Comput. Graph.*, 11(2):242–248, July 1977. ISSN 0097-8930.

M. Dhome, M. Richetin, J.-T. Laprest, and G. Rives. Determination of the Attitude of 3D Objects from a Single Perspective View. *IEEE Trans. Pattern Anal. Mach. Intell.*, 11(12):1265–1278, 1989.

A. DLR. Stereo Reconstruction (AT-RMPK-TN-001_1-1). Technical Report 1, Deutsches Zentrum für Luft- und Raumfahrt (DLR), 2011. internal document.

T. Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, pages 861–874, 2006.

P. D. Fiore. Efficient Linear Solution of Exterior Orientation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(2):140–148, 2001.

M. A. Fischler and R. C. Bolles. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Commun. ACM*, 24(6):381–395, June 1981. ISSN 0001-0782.

C. D. Ghilani. *Adjustment Computations: Spatial Data Analysis*. Wiley, 2011. ISBN 9781118174401.

E. Gülch. *Erzeugung digitaler Geländemodelle durch automatische Bildzuordnung*. Deutsche Geodätische Kommission bei der Bayerischen Akademie der Wissenschaften. Bayerische Akademie der Wissenschaften, 1994. ISBN 9783769694628.

R. M. Haralick, H. Joo, C. Lee, X. Zhuang, V. G. Vaidya, and M. B. Kim. Pose Estimation from Corresponding Point Data. *Systems, Man and Cybernetics, IEEE Transactions on*, 19(6):1426–1446, 1989.

R. M. Haralick, D. Lee, K. Ottenburg, and M. Nolle. Analysis and Solutions of The Three Point Perspective Pose Estimation Problem. In *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR '91., IEEE Computer Society Conference on*, pages 592–598, 1991.

R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, New York, NY, USA, 2 edition, 2003. ISBN 0521540518.

J. Haruyama, S. Hara, K. Hioki, A. Iwasaki, T. Morota, M. Ohtake, T. Matsunaga, H. Araki, K. Matsumoto, Y. Ishihara, H. Noda, S. Sasaki, S. Goossens, and T. Iwata. LUNAR GLOBAL DIGITAL TERRAIN MODEL DATASET PRODUCED FROM SELENE (KAGUYA) TERRAIN CAMERA STEREO OBSERVATIONS. *Lunar and Planetary Inst. Science Conf. Abstracts*, 43:1200, March 2012.

R. Horaud, B. Conio, O. Leboulleux, and B. Lacolle. An analytic solution for the perspective 4-point problem. *Computer Vision, Graphics, and Image Processing*, 47(1): 33–44, 1989.

B. Horn. *Robot Vision*. MIT electrical engineering and computer science series. MIT Press, 1986. ISBN 9780262081597.

M.-K. Hu. Visual Pattern Recognition by Moment Invariants. *Information Theory, IRE Transactions on*, 8(2):179–187, February 1962.

A. Huertas, Y. Cheng, and R. Madison. Passive Imaging Based Multi-cue Hazard Detection for Spacecraft Safe Landing. *Jet Propulsion Laboratory*, 2006.

A. E. Johnson and J. F. Montgomery. Overview of Terrain Relative Navigation Approaches for Precise Lunar Landing. *Proc. IEEE Aerospace Conference(Aerospace08),March 2008.*, 2008.

A. E. Johnson, A. Ansar, L. H. Matthies, N. Trawny, A. I. Mourikis, and S. I. Roumeliotis. A General Approach to Terrain Relative Navigation for Planetary Landing, 2007.

A. E. Johnson, A. Huertas, R. Werner, and J. F. Montgomery. Analysis of On-Board Hazard Detection and Avoidance for Safe Lunar Landing. *IEEE Aerospace Conference*, 2008.

J. N. Kapur, P. K. Sahoo, and A. K. Wong. A New Method for Gray-Level Picture Thresholding Using the Entropy of the Histogram. *Comput. Vision Graph. Image Process.*, 29:273–285, 1985.

V. Lepetit, F. Moreno-Noguer, and P. Fua. Accurate Non-Iterative O(n) Solution to the PnP Problem. *11th IEEE International Conference on Computer Vision*, 2007.

B. Li, Q. Meng, and H. Holstein. Point Pattern Matching and Applications-a Review. In *SMC*, pages 729–736. IEEE, 2003. ISBN 0-7803-7952-7.

C. C. Liebe. Pattern Recognition of Star Constellations for Spacecraft Applications. *IEEE Aerospace and Electronic Systems Magazine*, 8(1):33–39, Nov. 1993.

M. Lingenauber, T. Bodenmüller, J. Bartelsen, B. Maass, H. Krüger, C. Paproth, S. Kuß, and M. Suppa. RAPID MODELLING OF HIGH RESOLUTION MOON-LIKE TERRAIN MODELS FOR TESTING OF OPTICAL LOCALIZATION METHODS. *12th Symposium on Advanced Space Technologies in Robotics and Automation*, 2013.

Q. Long and L. Zhongdan. Linear N-Point Camera Pose Determination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1999.

D. G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. Comput. Vision*, 60(2):91–110, Nov. 2004. ISSN 0920-5691.

C.-P. Lu, G. D. Hager, and E. Mjolsness. Fast and Globally Convergent Pose Estimation from Video Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:610–622, 2000.

B. Maas and M. Verveld. ATON-Koordinatensystemreferenz(AT-RYNR-TN-001). Technical Report 10, Deutsches Zentrum für Luft- und Raumfahrt (DLR), 2011. internal document.

J. B. MacQueen. SOME METHODS FOR CLASSIFICATION AND ANALYSIS OF MULTIVARIATE OBSERVATIONS. In L. M. L. Cam and J. Neyman, editors, *Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297. University of California Press, 1967.

E. Mikhail and F. Ackermann. *Observations and least squares*. University Press of America, 1982.

Moon-C.S.C.E., S. Board, D. Sciences, and N. Council. *The Scientific Context for Exploration of the Moon: Final Report*. National Academies Press, 2007. ISBN 9780309109192.

A. Morfopolous, B. Metz, C. Villalpando, L. H. Matthies, and N. Serrano. Implementation of Pin Point Landing Vision Components in an FPGA system. In *Proceedings of the 2011 IEEE Aerospace Conference*, AERO '11, pages 1–9, Washington, DC, USA, 2011. IEEE Computer Society. ISBN 978-1-4244-7350-2.

NASA(JPL-photojournal). Landing accuracy on mars: A historical perspective. URL `http://photojournal.jpl.nasa.gov/catalog/PIA16039`. Last visit: 08.01.2014.

F. Neitzel. Ausgleichsrechnung - modellbildung, auswertung, qualittsbeurteilung. *Qualitätsmanagment geodätischer Mess- und Auswerteverfahren, Beiträge zum 93. DVW -Seminar am 10. und 11. Juni 2010. DVW-Schriftenreihe Band 61/2010*, 2010.

W. Niemeier. *Ausgleichungsrechnung: eine Einführung für Studierende und Praktiker des Vermessungs- und Geoinformationswesens*. De-Gruyter-Lehrbuch. Mathe, 2002. ISBN 9783110140804.

W. Niemeier. *Ausgleichsrechnung: statistische Auswertemethoden*. De-Gruyter-Lehrbuch. Bod Third Party Titels, 2008. ISBN 9783110190557.

G. Paar, F. Rottensteiner, and Pölzleitner. Image Matching Strategies. In *Digital Image Analysis: Selected Techniques and Applications*. Springer, New York, 2001. ISBN 0387950664.

C. Padgett, K. Kreutz-Delgado, and S. Udomkesmalee. Evaluation of Star Identification Techniques. *Journal of Guidance, Control, and Dynamics*, 20(2):259–267, 1997.

B.-V. Pham, M. Devy, S. Lacroix, M. Drieux, and C. Philippe. Visual Landmark Constellation matching for spacecraft pinpoint landing. In *AIAA Guidance, Navigation, and Control Conference*, 2009.

V. Rodehorst. *Photogrammetrische 3D-Rekonstruktion im Nahbereich durch Auto-Kalibrierung mit projektiver Geometrie*. Ph.d. dissertation, Berlin University of Technology, June 2004.

S. Suzuki and K. Abe. Topological structural analysis of digitized binary images by border following. *Computer Vision, Graphics, and Image Processing*, 30(1):32–46, 1985.

R. Szeliski. *Computer Vision: Algorithms and Applications*. Springer-Verlag New York, Inc., New York, NY, USA, 1st edition, 2010. ISBN 1848829345, 9781848829343.

S. Theil. Projektplan: ATON-Autonmous Terrain based Optical Navigation for Landers. Technical Report 1.3, Deutsches Zentrum für Luft- und Raumfahrt (DLR), 2011a. internal document.

S. Theil. Systemdefinition der ATON (AT-RYNR-TN-009_1-0). Technical Report 1, Deutsches Zentrum für Luft- und Raumfahrt (DLR), 2011b. internal document.

S. Theil and H. Krueger. Definition Referenzmissionen (AT-RYNR-TN-004_1-5). Technical Report 5, Deutsches Zentrum für Luft- und Raumfahrt (DLR), 2011. internal document.

N. Trawny, A. I. Mourikis, S. I. Roumeliotis, A. E. Johnson, and J. F. Montgomery. Vision-aided inertial navigation for pin-point landing using observations of mapped landmarks. *Journal of Field Robotics*, 24(5):357–378, May 2007. ISSN 1556-4959.

T. Tuytelaars and K. Mikolajczyk. *Local Invariant Feature Detectors: A Survey*. Now Publishers Inc., Hanover, MA, USA, 2008. ISBN 1601981384, 9781601981387.

B. Van Pham, S. Lacroix, and M. Devy. Vision-based absolute navigation for descent and landing. *Journal of Field Robotics*, 29(4):627–647, 2012. ISSN 1556-4967.

D. E. Walial and A. Suneja. A Conceptual Study on Image Matching. *Global Journal of Computer Science and Technology*, 10(12):83–88, Oct. 2010.

L. Williams. CASTING CURVED SHADOWS ON CURVED SURFACES. In *In Computer Graphics (SIGGRAPH 78 Proceedings*, pages 270–274, 1978.

A. Woo, P. Poulin, and A. Fournier. A Survey of Shadow Algorithms. *IEEE Computer Graphics and Applications*, 10:13–32, 1990.

M. J. Zaki and W. Meira. *Data Mining and Analysis: Fundamental Concepts and Algorithms*. Cambridge University Press, Feb. 2013. ISBN 0521766338.

D. Zhang and G. Lu. Review of shape representation and description techniques. *Pattern Recognition*, 37:1–19, 1 2004.